THE UNIVERSITY OF ALBERTA

DATA ACQUISITION SYSTEM FOR BIOTELEMETRY

by

Ⓒ LAURIE J. DAVIS

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES AND RESEARCH
IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE
OF MASTER OF SCIENCE

DEPARTMENT OF ELECTRICAL ENGINEERING

EDMONTON, ALBERTA
FALL, 1978

THE UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research, for acceptance, a thesis entitled DATA ACQUISITION SYSTEM FOR BIOTELEMETRY

submitted by LAURIE J. DAVIS

in partial fulfilment of the requirements for the degree of

MASTER OF SCIENCE

Pat Harding

Supervisors

Keith C.

Wm A Fuller

Date Sept. 28 /78

Abstract

This thesis describes the design of a data acquisition system used for tracking the movements of small mammals. An orthogonal grid of beverage antennas is used to receive signals from transmitters which have been placed on the animals. By reading the signal strengths on each antenna, it is possible to determine the coordinates of the transmitter. By utilizing transmitters at different frequencies, it is possible to track several animals simultaneously.

The function of the data acquisition system is to periodically scan the antenna system, determine the position of each animal, and display the coordinates. Bulk storage of data is provided by a cassette tape system.

The use of a microcomputer to control the actions of the system greatly enhances its capabilities. It is possible to interpolate between the signal readings in order to increase the resolution of the system. It is also possible to make decisions as to which data are relevant and should be stored. The major advantage of this system is the mass storage facility. This facility allows the data to be processed at a later time without manually reentering it.

Acknowledgements

The author wishes to express his appreciation to his supervising professors, Dr. F.S. Chute and Mr. P.J.R. Harding for their help throughout the course of this work.

Sincere thanks are also extended to the staff of the Department of Electrical Engineering, and to the other graduate students within the department, especially Michael Weiss and Martin Nohr.

The financial support provided by the National Research Council of Canada and the University of Alberta Department of Zoology through Dr. W.A. Fuller is also gratefully acknowledged.

Table of Contents

- INTRODUCTION........1
- SYSTEM DESCRIPTION........5
  - Transmitter........5
  - Antenna System........5
  - Receiver........7
  - Microcomputer System........8
- HARDWARE DESCRIPTION........14
  - Antenna Control........14
  - Radio Receiver........17
  - Microcomputer........23
    - Central Processor Card........23
    - Program Memory........28
    - Read/Write Memory........29
    - Keyboard/Printer Control Card........31
    - Cassette Interface Card........36
    - Antenna/Radio Control Card........43
    - Power Supplies........47
- SOFTWARE DESCRIPTION........49
    - Initialization........51
    - Main Program........51
    - Keyboard Routines........52
    - System Monitor........53
    - Display Routines........54
    - Utility Subroutines........55
    - Stepper Motor Control Programs........57

Printer Control Programs..........................58
Antenna Control Programs..........................60
Hardware Timer Delay..............................61
CB Radio Control Programs.........................61
Interrupt Driven Printer Routine..................62
Interrupt Driven Printer Subroutines..............64
Real Time Clock Subroutines.......................64
Scan Subroutines..................................65
Interrupt Routine for Cassette Interface..........66
Cassette Interface Subroutines....................67
Real Time Clock Interrupt Routine.................69
Scan Subroutines..................................76
SUMMARY...86
REFERENCES..88
APPENDIX A: User's Guide..........................90
APPENDIX B: Circuit Diagrams......................112
APPENDIX C: Program Listing.......................120

List of Figures

Figure Page

1. Transmitter circuit....................................6
2. System Diagram...9
3. Block diagram..10
4. Memory Map...11
5. Control Signals to Antenna Box.........................16
6. Receiver Circuit Diagram...............................18
7. Modifications to Receiver..............................19
8. Control Signals to Radio...............................23
9. Converting NRZ to Phase Encoded Data...................42
10. Real Time Clock Interrupt Routine.....................71
11. Scan Routine..73
12. Relay Control Card....................................112
13. Central Processor Card................................113
14. Program Memory Card...................................114
15. Read/Write Memory Card................................115
16. Keyboard/Printer Control Card.........................116
17. Cassette Interface Card...............................117
18. Antenna/Radio Control Card............................118
19. Power Supplies..119

I. INTRODUCTION

Many methods have been devised for tracking small mammals. The movement of the animals can be traced by the tracks they leave in sand (Bider 1968), or on sheets of smoked paper (Justice 1961). They have also been tracked by following a trail of radioactive excretory products (Miller 1957). Other methods which have been employed include automatic cameras, and microswitches on swinging doors. The earliest attempts at radio tracking used a system of highly directional receiving antennas on high towers. The transmitter position was then determined by triangulation.

All methods of tracking animals have some disadvantages. The intrusion of man into the natural habitat of the animals upsets the normal life cycles of the area. For this reason, the amount of direct contact by man should be kept to a minimum. Some type of remote sensing system is therefore suggested. The devices used to do the tracking should disturb the animals as little as possible. For example, if radio transmitters are used, they must be kept small enough and light enough that they do nct hinder the movement of the animals.

A recent attempt has been made to trace the movements of these animals by using a grid of overhead wire antennas (Chute et al. 1974). Small transmitters are attached to the animals, which are then released under the antenna system. The purpose of the present project is to design and build a microprocessor controlled system to monitor these antennas and determine the positions of the transmitters under the array. The antenna system is already in use by the Department of Zoology at the University of Alberta. It consists of an orthogonal grid of 42 horizontal wire antennas, 21 in each direction. One end of each antenna is connected, via coaxial cable, to the antenna control system.

In the system described in the following sections, the antennas are systematically connected, under computer control, to a receiver where the signal strengths are measured. After the signal strengths from all of the antennas have been recorded, the computer processes the data to determine the location of the transmitter. The location is indicated as the intersection of the two orthogonal wire antennas providing the largest signals to the receiver.

The processed data are then displayed on a printer, and/or stored on a cassette tape. The printer is used to provide the user with immediate results. The data are printed as they are collected. The cassette tape system provides bulk storage of data. The data are stored on tape, in machine readable form, so that they can be processed later by a general purpose computer.

Besides recording the position of each transmitter, the computer determines the activity state of each transmitter. There are two activity states that the system can recognize. A transmitter is said to be in the first activity state if it is motionless, while the second activity state indicates a transmitter in motion. The motion can be of two types. The simplest to detect is a transmitter which is moving from one location to another. The transmitter can, however, move without changing location. An example of this would be an animal which is in one location, but moving about in that location (e.g. grooming itself). The activity state is indicated by a flag in the output to the printer or cassette tape. A flag is a space reserved in the output which can contain either a zero or a one. In this case, a zero indicates a transmitter which is motionless, while a one indicates a transmitter in motion.

A test antenna is also provided to monitor the level of radio interference in the area. The test antenna is similiar to individual antennas in the grid, but is placed outside the grid. The test antenna is located far enough away from the grid that transmitters placed under the grid will not affect it. Any signals picked up on the test antenna can then be attributed to external sources, which may affect the signal levels on the grid. The computer checks the test antenna each time the antenna system is scanned. If the signal level on a particular channel is too high, a flag is set in the output to indicate that the data may be in error. This flag is set to a one if the signal on the test antenna is greater than a particular level.

A facility is also provided which allows the user to keep track of the number of times a transmitter is found at a particular location. In order to implement this population matrix facility, a set of counters is kept in the computer memory, one counter for each possible location. Whenever a transmitter position is determined, the counter corresponding to that position is incremented. These data are printed upon receipt of the appropriate command from the user.

The system incorporates a typewriter-like keyboard similiar to a standard computer terminal. This keyboard allows commands to be issued to the computer so that the user has complete control over the operation of the system.

II. SYSTEM DESCRIPTION

A. Transmitter

The transmitter consists of a crystal oscillator operating in the citizens band, and a transmitting antenna. A circuit diagram of the transmitter is given in figure 1. Inductor L2 is used as a collar as well as the transmitting antenna. Since L2 is small compared to L1, the tuned circuit frequency is not sensitive to the collar size. The output signal from the transmitter is sufficient to couple into the antenna system since it is only a short distance away.

B. Antenna System

The antenna system consists of an orthogonal grid of horizontal wire antennas, 21 in each direction. Each antenna consists of a 30 meter length of wire, spanned one meter above the ground. One end of the antenna wire is terminated in a ground connection, while the other end is connected, via coaxial cable, to the antenna control system. The coaxial cable connects each antenna to a relay in the antenna control box (see figure 2). The other side of the relay is connected to the input of the radio. In order to select a particular antenna, a code word is sent from the computer to the antenna control system. The logic in the antenna control system decodes the control signals and

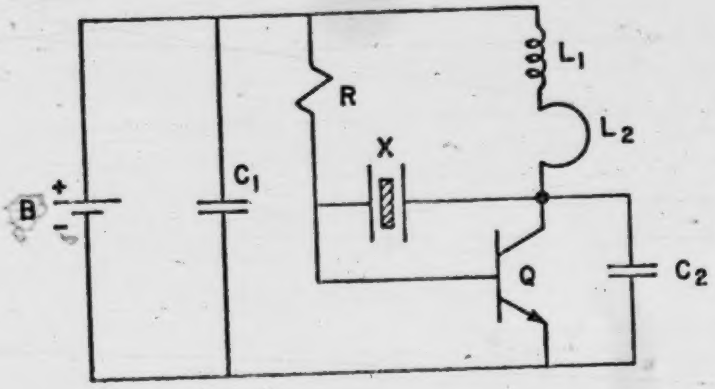


Fig. 1 Transmitter circuit

activates the appropriate relay. In this way the computer can connect any antenna to the input of the radio.

The first step in scanning the antenna system is to connect a particular antenna to the radio. The computer then sets the radio to the channel to be monitored. The signal strength on the antenna is then read and stored in memory. If there are transmitters on other channels, the radio is then set to one of the other channels and the signal strength for that channel is recorded. This procedure continues until all channels have been monitored. At this point the computer disconnects the antenna and connects the next antenna to be checked. All channels are again monitored and the readings are stored in memory. After the entire antenna system has been scanned in this manner, the data stored in memory are processed to determine relative signal strengths and hence the coordinates of each transmitter.

Since the antennas closest to the transmitter receive the strongest signal, the transmitter position can be determined by comparing the signal strengths on all of the antennas. The data from the two sets of antennas (X and Y) are processed separately so that two orthogonal coordinates can be determined.

C. Receiver

The receiver used is a standard citizens band transceiver. A CB system was chosen because of the low cost and availability of the crystals for the transmitters, as well as the availability of commercial receivers. The use of the CB band is viable in this case because of the location of the system. If the system were to operate in an area with a large amount of CB activity, a different frequency band may have to be chosen. In order to allow computer control of the receiver channel, the channel selection system in the original receiver had to be modified. The radio was rewired so that the channel selection circuitry received its control signals from the computer, rather than from the conventional rotary switch. For proper system operation the radio must supply the computer with a signal proportional to the received signal strength. Modifications to the radio were required to make this signal available. These changes will be detailed in a following section.

D. Microcomputer System

The purpose of the microprocessor system is to monitor the antenna system, determine the location of each transmitter within the grid, and store this information on a cassette tape so that further processing is possible at a later date by a general purpose computer.

The system is controlled by a microcomputer which consists of a central processing unit and its associated memory. The function of the central processor is to read program instructions from memory, decode the instruction, and finally perform whatever task the instruction requested. Memory is divided to two major categories. Program memory contains the control program for the system. All operations performed by the computer are initiated by instructions read from the program memory. The second section of memory provides for temporary storage of data. For example, this section of memory contains the raw data received from the antennas, the processed data waiting to be printed, and the counters for the population matrix facility. The memory map in figure 4 gives a breakdown of memory usage.

In order to allow processing of the data by a general purpose computer, the data collected are stored on a cassette tape. Data stored on the tape include the real time, the channel number, the coordinates of the particular transmitter, the activity state of the animal, and the condition of the test antenna. In order to conserve space on the cassette tape, not all of the data collected are stored

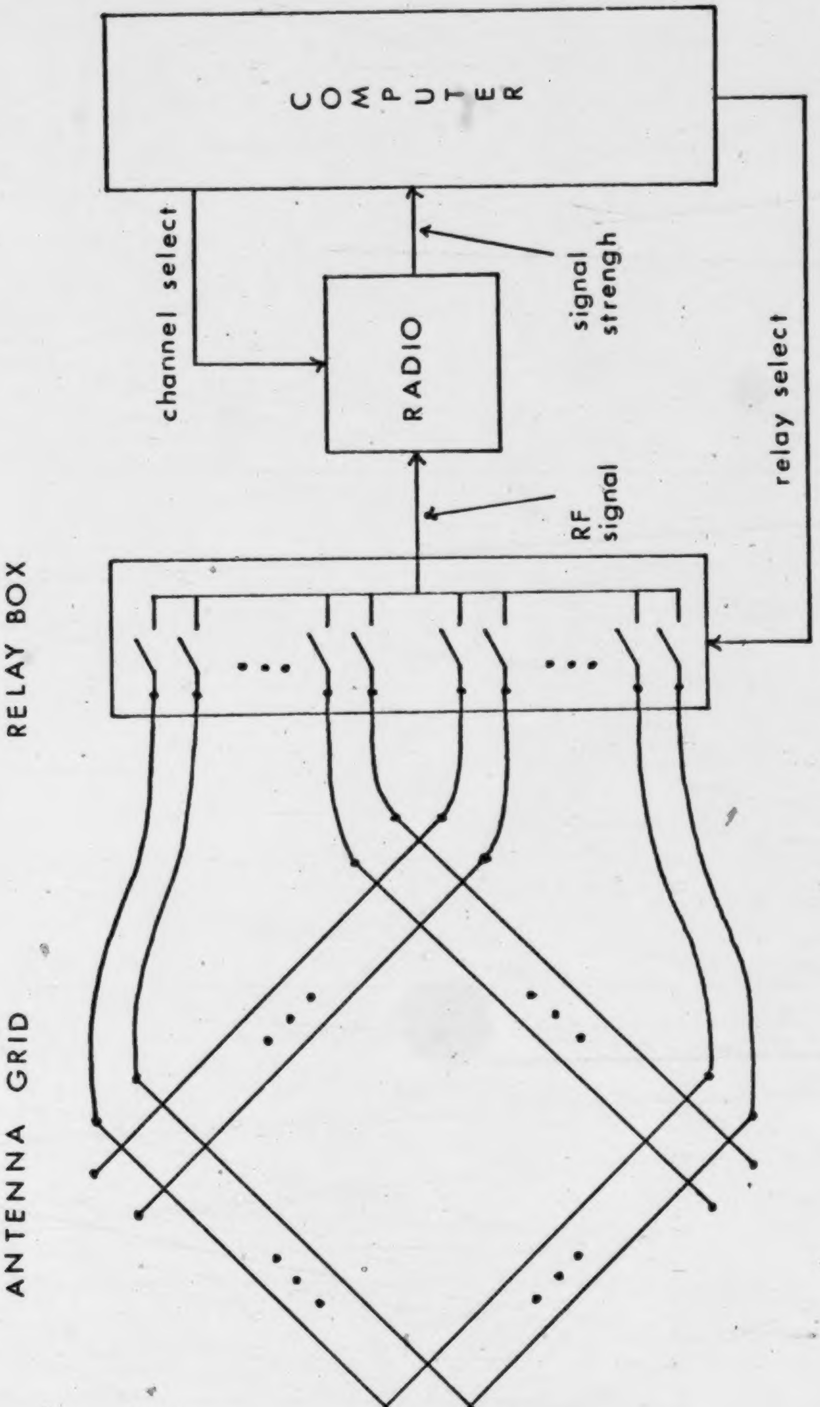

COMPUTER
channel select
signal strengh
RADIO
relay select
RF signal
RELAY BOX
ANTENNA GRID

Fig. 2 System Diagram

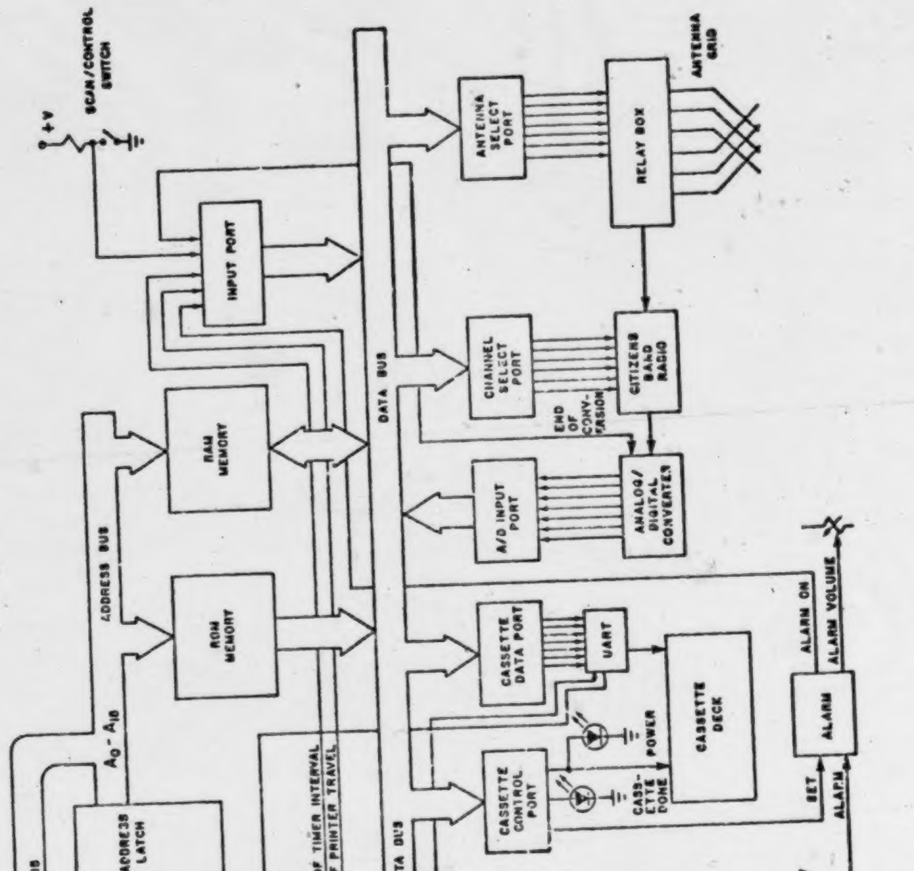

SCAN/CONTROL SWITCH
+V
INPUT PORT
ANTENNA SELECT PORT
RELAY BOX
ANTENNA GRID
CHANNEL SELECT PORT
CITIZENS BAND RADIO
END OF CONVERSION
DATA BUS
RAM MEMORY
ADDRESS BUS
A/D INPUT PORT
ANALOG/DIGITAL CONVERTER
ALARM ON
ALARM VOLUME
ROM MEMORY
CASSETTE DATA PORT
UART
CASSETTE DECK
POWER
CASSETTE DONE
CASSETTE CONTROL PORT
ALARM
SET
ALARM
$A_0 - A_{10}$
ADDRESS LATCH
$A_8 - A_{15}$
$AD_0 - AD_7$
END OF TIMER INTERVAL
END OF PRINTER TRAVEL
DATA BUS
PRINTER DATA PORT
THERMAL PRINTER
+V
CASSETTE RESET SWITCH
MICRO-PROCESSOR
RST 6.5
RESET
RST 7.5
RST 5.5
PRINTER MOTOR CONTROL PORT
REAL TIME CLOCK
+V
RESET
MASTER RESET
TIMER
PRINT STROBE
CASSETTE BAUD RATE
DATA BUS
KEYBOARD PORT
KEYBOARD

Fig. 3 Block diagram

	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	ADDRESS SPACE
ROM 1	0	0	0	0	0	0	X	X	X	X	X	X	X	X	X	X	0000-03FF
ROM 2	0	0	0	0	0	1	X	X	X	X	X	X	X	X	X	X	0400-07FF
ROM 3	0	0	0	0	1	0	X	X	X	X	X	X	X	X	X	X	0800-0BFF
ROM 4	0	0	0	0	1	1	X	X	X	X	X	X	X	X	X	X	0C00-0FFF
ROM 5	0	0	0	1	0	0	X	X	X	X	X	X	X	X	X	X	1000-13FF
ROM 6	0	0	0	1	0	1	X	X	X	X	X	X	X	X	X	X	1400-17FF
ROM 7	0	0	0	1	1	0	X	X	X	X	X	X	X	X	X	X	1800-1BFF
ROM 8	0	0	0	1	1	1	X	X	X	X	X	X	X	X	X	X	1C00-1FFF
RAM 1	0	0	1	0	0	0	X	X	X	X	X	X	X	X	X	X	2000-23FF
RAM 2	0	0	1	0	0	1	X	X	X	X	X	X	X	X	X	X	2400-27FF
RAM 3	0	0	1	0	1	0	X	X	X	X	X	X	X	X	X	X	2800-2BFF
RAM 4	0	0	1	0	1	1	X	X	X	X	X	X	X	X	X	X	2C00-2FFF
4K CARD	0	0	1	1	X	X	X	X	X	X	X	X	X	X	X	X	3000-3FFF

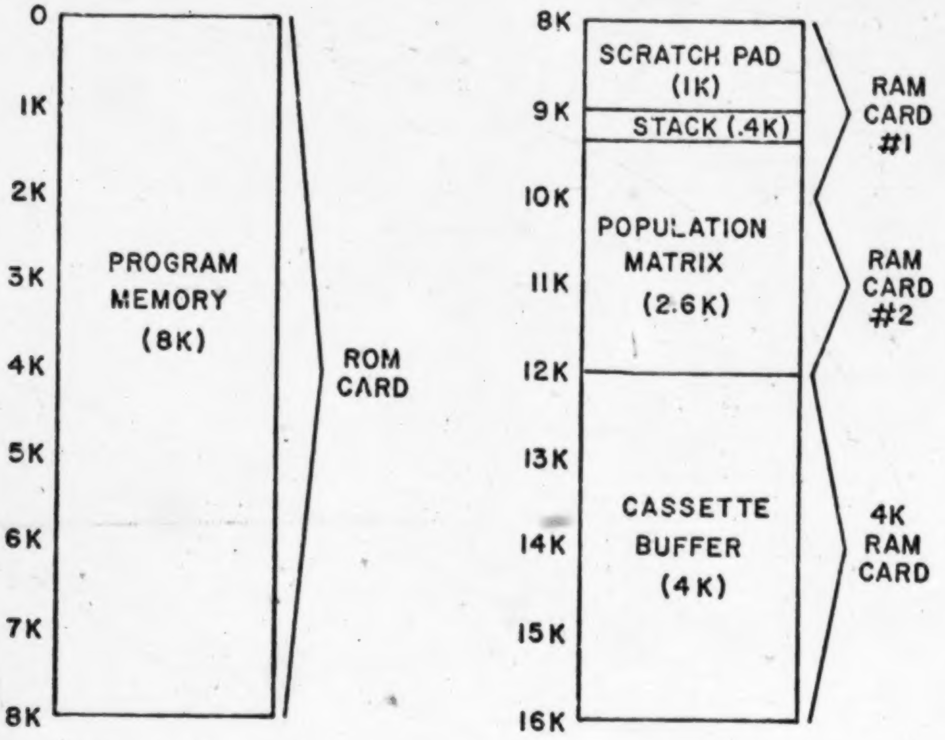

0
1K
2K
3K
4K
5K
6K
7K
8K
PROGRAM MEMORY (8K)
ROM CARD
8K
9K
10K
11K
12K
13K
14K
15K
16K
SCRATCH PAD (1K)
STACK (.4K)
POPULATION MATRIX (2.6K)
CASSETTE BUFFER (4K)
RAM CARD #1
RAM CARD #2
4K RAM CARD

Fig. 4 Memory Map

on the tape. Each time data are collected, they are compared to the previous set of data. If nothing has changed, no information is contained in the new data, so they are not stored. One result of this method of storing data is a very irregular rate of data flow to the cassette tape system. If the data were loaded directly onto the tape, there would be large gaps between sets of data. In order to overcome this problem, the data are not sent directly to the cassette tape, but are stored in a buffer memory. Data are stored in the buffer until the buffer becomes full. At this point, the system automatically turns on the cassette drive, and transfers all of the data in the buffer onto the cassette tape. The cassette drive is then turned off. In this way, whenever the cassette drive is on, data are sent to the tape at the maximum possible rate, thus conserving tape space. Whenever the tape becomes full, an alarm is sounded. Since there is a period of time when the cassette drive is off (when the buffer is filling up), a new tape can be inserted without interrupting the system.

Operator control over the system is provided by a keyboard. Before each experiment, various parameters must be entered into the system. These include such items as the correct time, the channels which are used, the size of the cassette tape, the scanning interval, etc. Once the system is initialized via the keyboard, it can collect data completely unattended. The only operator intervention required is the replacing of the cassette tapes as they become full.

A block diagram of the system is given in figure 3. The top half of the diagram is the standard microcomputer configuration. The bus oriented system allows the microcomputer to control various peripheral equipment such as the keyboard, printer, etc. Physically the computer occupies five cards on the backplane. All other cards, when inserted into the backplane, have access to the address, data, and control buses. Because of this modular arrangement, any additions to the system can be made without altering the existing system. Any lines which connect to external devices (such as the radio) are connected through the backplane. This arrangement allows the cards to be removed without disconnecting any cables.

The computer and the associated power supplies are housed in one cabinet along with the keyboard, printer, and cassette drive. The radio and antenna control box are housed in another cabinet. In order to reduce the amount of system noise picked up by the radio, these two cabinets are kept as far apart as possible.

III. HARDWARE DESCRIPTION

The hardware consists of three major sections. The antenna control box controls the antenna system. Under control of the computer, a particular antenna can be connected to the radio. The radio receives the signal from the antennas, sorts out the proper channel, and measures the signal strength. The signal strength is then read by the computer which converts it to a digital code. The computer then analyses the data and determines the location of each transmitter.

A. Antenna Control

The computer selects a particular antenna by sending appropriate control signals to the antenna control system. These signals are then decoded and the correct relay is activated. The control inputs to the antenna control box consist of 5 signals which are used to select a particular antenna, plus a signal ($X/\overline{Y}$) which determines which set of antennas (X or Y) will be selected.

The relays for the two sets of antennas (X and Y) are each on a separate board. Figure 12 is the circuit diagram for the relay cards. The boards are identical, but have jumper options to enable the card to be configured either as the X or Y card. The circuitry is basically a two level decoder. The first level is a 2 to 4 decoder (74LS138). A 2 bit binary input results in the selection of one of the 4 outputs. Each binary input has a unique output. Three of the outputs of this stage select one of three 3 to 8 decoders (74156). The entire configuration can be considered as a 5 to 24 decoder. A 5 bit binary input results in the selection of one of the 24 outputs. The first 22 outputs are used to turn on the relay corresponding to the binary input.

Card selection is accomplished by changing the enable signals to the first level of the decoder. Whenever the first level of the decoder is disabled, none of its outputs are selected. Since these outputs are used to enable the second level of the decoder, none of the 24 cutputs are selected. For the X card, the active high enable (G1) is connected to $X/\overline{Y}$, while the active low enable is grounded. On the Y card, the active low enable (G2A) is connected to $X/\overline{Y}$, while the active high enable is pulled high. Therefore a high signal on $X/\overline{Y}$ selects the X card, while a low signal selects the Y card.

The outputs of the second level of the decoder directly drive the reed relay coils. The outputs are open collector, with enough sink capability (16 mA) to turn on the relay. Since the outputs are active low, one end of the relay coil is tied to +5 volts, and the other end is connected to the decoder output. When a particular output is selected, the output goes low, turning on the relay. The relays used to connect the antennas to the radio are CLARE PRME 1A005 reed relays. These are five volt relays which require approximately 2 milliseconds to close.

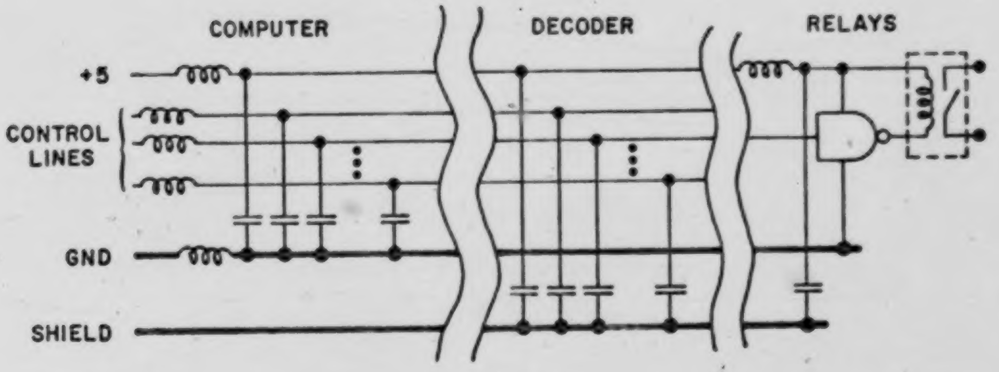


Fig. 5 Control Signals to Antenna Box

One side of the relay contacts is connected to an individual antenna, while the other side is connected to the radio, in parallel with all the other relays. When one relay is selected, the appropriate antenna is therefore connected to the radio.

The major problem encountered in the operation this section was the noise picked up on the signal lines and fed into the radio. It was discovered that the major source of this noise was associated with the generation of the control signals in the computer. In order to reduce this noise, the filtering and decoupling scheme of figure 5 was used.

Besides decoupling the control lines, care had to be taken to completely shield the system. The two cards are completely enclosed in a metal box, with the RF signals entering and leaving through BNC connectors. The control signals from the computer are brought in with a shielded cable, with the shield connected to the shield box (see figure 5). The shielding philosophy was to think of the relay box as an extension of the coaxial cable from the antennas to the radio. Therefore the shield on the control cable is not connected to the computer ground, but to the shield box.

B. Radio Receiver

The radio receiver used is a Radio Shack model TRC-456. This transceiver is a Citizens Band radio, with phase locked loop tuning which provides 40 individual channels. The input sensitivity is specified as 0.2 uV for 10 dB S/N, and the selectivity is ±3 KHz at -6dB.

Several modifications were required before the receiver could be used in this application. The channel selection had to be controlled by the computer, and an output had to be provided which was proportional to the signal strength. Since it was also discovered that the radio was incapable of responding fast enough to a channel change, modifications were required to speed up the response. Modifications to the radio are shown in figures 6 and 7.

Frequency synthesis in the radio is done with a phase locked loop system. IC801 contains most of the circuitry of a standard phase locked loop. The programmable divider in the feedback loop is the key to controlling the channel selection. The binary inputs to the divider determine the

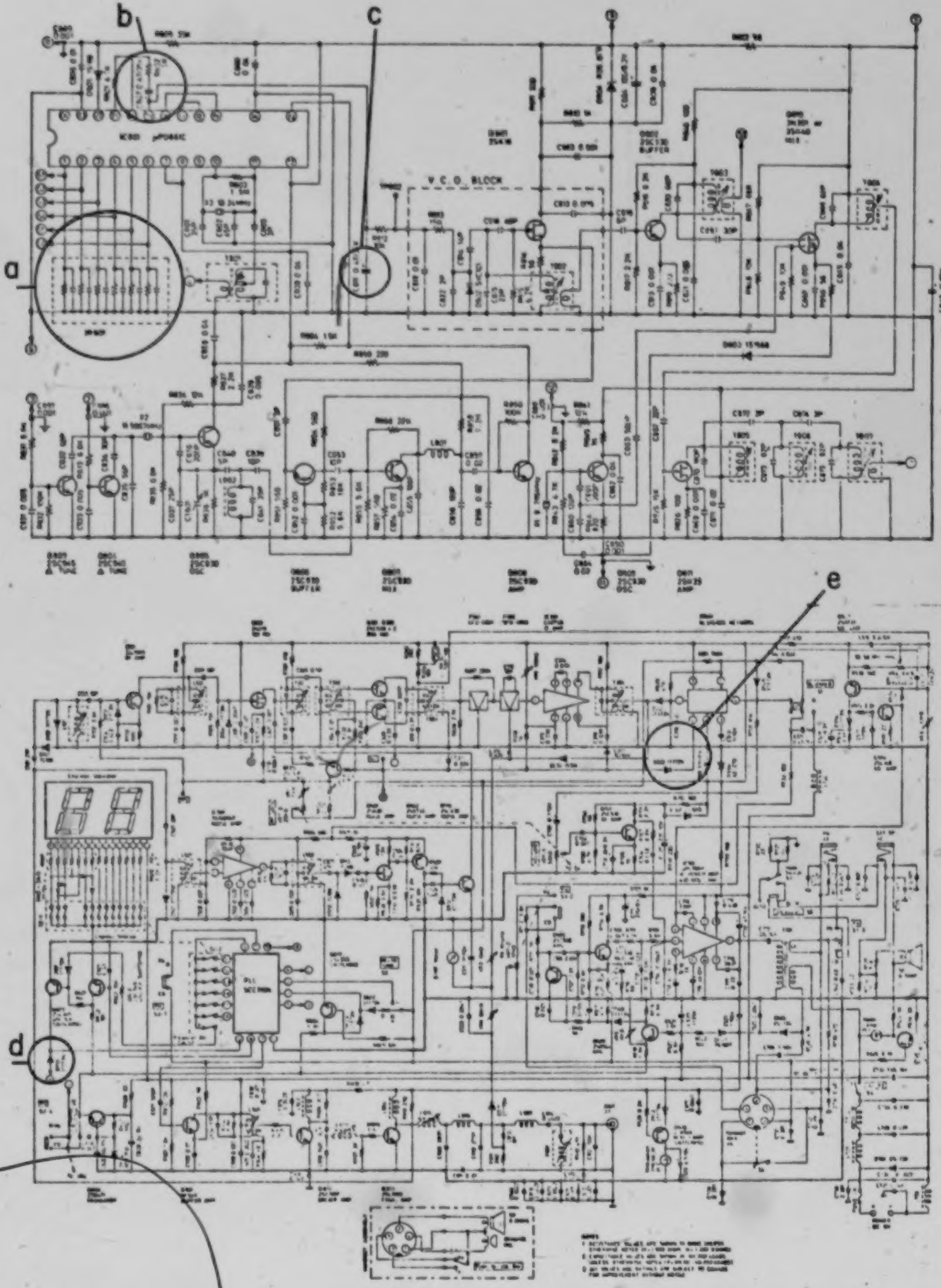

b
c
a
V.C.O. BLOCK
e
d

Fig. 6 Receiver Circuit Diagram

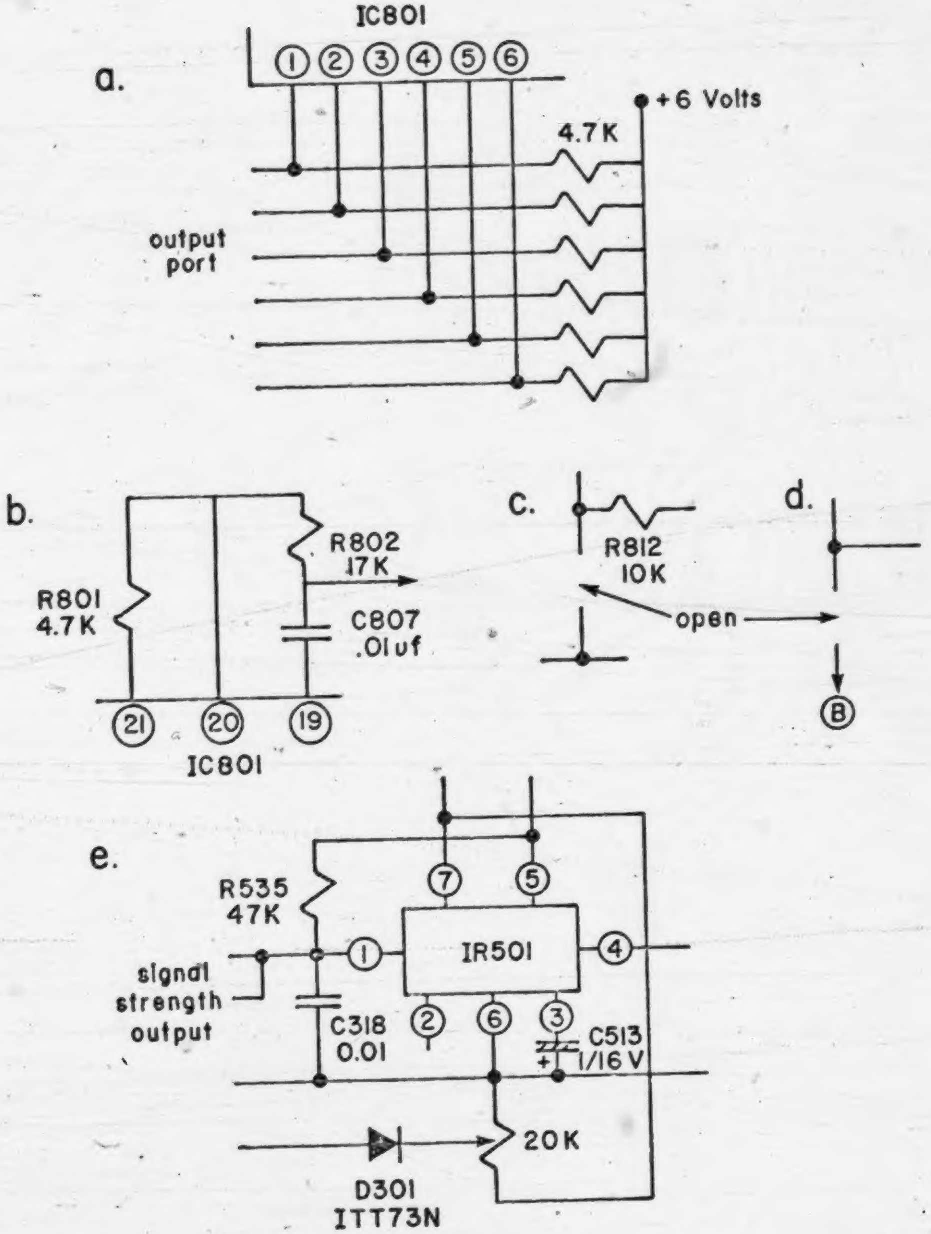

a.
IC801
1
2
3
4
5
6
+6 Volts
4.7K
output port
b.
R801
4.7K
R802
17K
C807
.01uf
21
20
19
IC801
c.
R812
10K
open
d.
B
e.
R535
47K
7
5
1
IR501
4
signal strength output
C318
0.01
2
6
3
C513
1/16 V
+
20K
D301
ITT73N

Fig. 7 Modifications to Receiver

channel selected. All that is required therefore, is to supply the appropriate binary code to IC801.

The receiver used a multi-wafer rotary switch to generate the appropriate code for the programmable divider. The rotary switch was disconnected and, in its place, an output port from the computer was installed (see figure 7a). In order to arrange for the proper logic levels (0 to 6 volts), the output port was buffered with optical isolators with the outputs pulled up to 6 volts. The radio channel is therefore selected by sending the appropriate binary word to the output port.

The computer must also be able to read the amplitude of the input signal. This information is easily obtainable from the radio. The automatic gain control circuit monitors the signal strength out of the IF amplifier and, based on the strength, sends a control signal to the RF amplifier to control its gain. The input to the AGC circuit is therefore proportional to the input signal strength. This is the signal that is sent to the computer for processing (see figure 7e).

Ideally the automatic gain control circuit keeps the signal strength out of the IF amplifier constant. In order to defeat this circuit, the output from the AGC circuit was disconnected (see figure 7e). Disconnecting the automatic gain control allows the output of the IF amplifier to follow the input signal strength.

It was found that simply disconnecting the AGC line caused the RF gain to drop undesireably low. It was necessary therefore, to control the level on the AGC line to set the RF amplifier to its maximum gain. A 20K potentiometer was connected between power and ground. The wiper was then connected to the AGC line and adjusted for maximum gain in the RF amplifier.

In order to scan the antenna system, it is necessary to take a signal strength measurement approximately every 3 msec. It was found that the radio could not change channels that fast. The first modifications (figures 7b and 7c) involved changing the filter in the phase locked loop. The cutoff frequency of the low pass filter had to be increased to allow the loop to respond faster. The cutoff frequency was increased by removing C811, and decreasing the value of C807. In order to maintain the same gain around the loop, R802 was increased in value. Before these changes, the receiver required 150 msec to switch between adjacent channels. After the modifications, the time was reduced to 2 msec.

These modifications allowed, however, only switching between adjacent channels. In order to increase the response time even further, it was necessary to defeat the 'unlock switching' circuit. It was found that the circuit could be defeated by removing diode D610 (see figure 7d). The radio then required a maximum of 2.5 msec to switch between any two channels. When the maximum number of channels (5) is scanned, the length of time required to monitor one antenna is therefore 12.5 msec. The entire antenna system can then be scanned in under one second (43 times 12.5 msec = 537.5 msec).

As with the antenna control box, it was found that a great deal of noise entered the system through the control lines from the computer. The circuit of figure 8 is used to reduce the level of noise entering the system. The inputs and outputs of the optical isolators are completely isolated. Complete isolation is made possible by providing the radio with its own isolated power supply. As discussed earlier, the optical isolators provide level shifting as well as isolation.

As was found with the antenna box, shielding proved to be an important consideration. Care had to be taken to ensure that the receiver was completely shielded from external noise. Again, since the radio should be considered an extension of the RF signal line, the shield on the control cable was not connected to the computer ground.

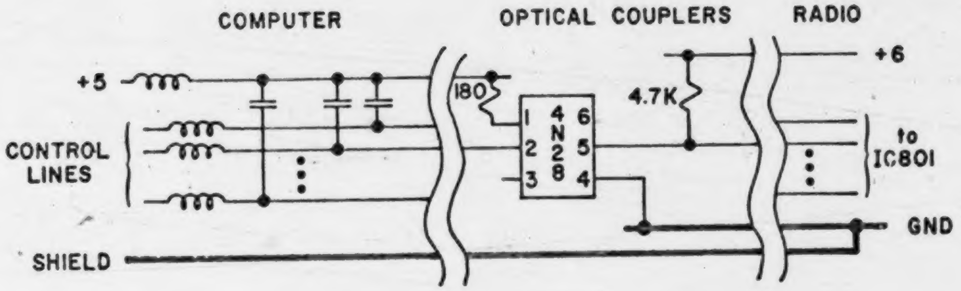

COMPUTER
OPTICAL COUPLERS
RADIO
+5
+6
180
4.7K
1
2
3
4N28
6
5
4
CONTROL LINES
to IC801
GND
SHIELD

Fig. 8 Control Signals to Radio

C. Microcomputer

Central Processor Card

The central processor card contains an 8085 CPU, the reset circuitry, the real time clock, and the bus buffers. This card is the main functional unit of the computer which controls the operation of the entire system. A circuit diagram of this card is given in figure 13.

8085 Processor

The Intel 8085 is a complete 8 bit central processing unit. All external hardware is controlled via the address bus, the multiplexed address/data bus, and the control bus. The CPU has provision for on-chip vectored interrupts as well as the system reset.

The CPU also contains the circuitry necessary to produce the system clock. The only external hardware required is a single crystal. Since internal circuitry divides down the crystal frequency, the 6 MHz crystal used results in a 3 MHz system clock. Since the system clock is also used as a real time clock, the frequency of the crystal was trimmed by adding capacitors in parallel.

Three of the interrupts provided on the 8085 (RST 5.5, RST 6.5, and RST 7.5) are used in this application. Whenever a high signal is detected on the pin corresponding to one of these interrupt lines, the CPU performs a jump to a location in memory particular to that interrupt. The program has, at that location, a jump to an interrupt service subroutine, which performs whatever tasks are necessary to satisfy the source of the interrupt.

Reset Circuitry

The reset circuitry provides a method of controlling the start-up mode of the CPU. Whenever a low signal is detected on the reset line, the processor jumps to the start of program memory. Any routines necessary to initialize the system are stored in this memory area so they are excecuted whenever the reset button is depressed. Provision is also made for a power-on reset. The capacitor (see figure 13) connected to the reset line keeps the reset line low when power is first applied. The capacitor then slowly charges through the 15K resistor until it reaches the logic '1' level of the CPU. At this point the processor can begin excecuting the program stored in the bottom of memory.

Real Time Clock

The real time clock circuitry consists of a series of programmable dividers which divide down the system clock from 3 MHz to 1 Hz. The 1Hz signal is then used to set a flip flop (74112), the output of which interrupts the CPU. The interrupt service subroutine for this particular interrupt (RST 6.5) then increments the clock which is contained in read/write memory. The interrupt service subroutine also resets the flip flop using the serial output port (SOD). As well as providing the real time clock signal, the series of dividers also has an intermediate output (2400 Hz) which is used to provide the timing for the cassette interface.

Bus Buffers

A large number of peripheral devices are connected to the central processor via the address, data, and control buses. In order to allow the central processor to drive these buses, buffers are provided to increase the drive capability. All buffers are inverting, but this does not present a problem since each peripheral device is equipped with receiving buffers which are also inverting.

The 8085 CPU uses a multiplexed data bus. During the first portion of an instruction, this bus contains the lower byte of the address. The combination of this bus with the address bus, which contains the upper byte of the address, can be used to select any address in the memory space. Whenever the lower byte of the address is valid on the multiplexed address/data bus, the ALE (address latch enable) signal goes high. This signal is used to latch the lower byte of the address. The bus is then free to be used as a data bus to transfer data to or from memory or peripheral hardware. In this application, an 8212 port/latch is used to latch the address. The output of the 8212 provides the lower address byte to the address bus on the backplane. The 8212 is enabled with the ALE signal.

Buffering the data bus is not as straight forward as buffering the address bus. First, the data bus is bi-directional, data must be able to move both to and from the CPU. Second, there must be provision to put the buffers in a high impedance state, so that other devices can obtain control of the bus. The data bus buffering was accomplished with 74366/368 bus buffers. These are inverting buffers which have an enable input. When the buffers are disabled, the outputs are in a high impedance state.

The input buffers are enabled with the CPU read (RD) signal. The result is that data are presented to the CPU only when it is ready to read the bus. The output buffers must be enabled under two conditions. First, whenever the CPU wishes to write data to the bus. In this case, the buffers are enabled with a write signal. Instead of using the WR signal, the status pin S1 is used. The S1 signal can be thought of as an advance write signal. The advantage of using this signal is that the buffers are enabled slightly before the CPU does a write to the data bus. Since there are several levels of buffering on the data bus, enabling the buffers in advance allows time for all of the buffers to turn on before the data transfer begins. The second condition for which the output buffers should be enabled is when there is a valid address on the bus. Since the 8212 address latch derives its input from the output of these buffers, the buffers must be enabled so that the address latch can read the address. For these reasons the output bus buffers are enabled with S1 (write) OR

ALE (address latch enable).

Program Memory

The program memory card contains a series of 2708 read only memories which contain the control program. The CPU reads all of its instructions from this card. The 2708 EPROM (erasable-programmable read only memory) is a 1024 by 8 bit memory which can be erased by exposure to ultraviolet light.

Since there are 1024 memory addresses, 10 bits of the address bus are used to select the proper memory locations. Since the address bus on the backplane is inverted, the lower 10 address bits are connected to the memories through 7404 inverting buffers. The program memory should be located at the bottom of memory, and since there are 8 EPROMs on the card, the addresses used are 0000 to 1FFF (see figure 14).

From the memory map it can be seen that in order to fully decode the memory, the program memory must be enabled when the highest three bits (A15, A14, and A13) are low. The decoding is accomplished by ANDing together ¬A15, ¬A14, and ¬A13 (see figure 14). This signal is then used to enable the decoder and the bus buffers. The next three bits (A12, A11, and A10) are used to select one of the eight EPROMs. A 3 to 8 decoder (74LS138) is used for this purpose. The three address lines are used as select inputs, with the active low outputs used as chip select signals to the individual ROMs. Since all address bits are now used in selecting a particular memory location, the address space is fully decoded.

Since the 8085 uses input/output instructions for I/O to external hardware, there is an overlap on the address bus between memory and I/O locations. For this reason, the memory and data bus buffer enables are also conditioned with the IO/M (input-output / memory) line. The memory is now selected only when the CPU is accessing memory. A read signal (S1) is also used to enable the memories since the 2708s are read only memories.

The data bus on this card is not bi-directional since the program memory card can only send data to the CPU. As can be seen from figure 14 , the data bus buffers are enabled with ADDRESS SELECT AND M (memory select) AND RD (read). Enabling the buffers in this manner ensures that the data bus buffers are in the high impedance state whenever the CPU is not accessing data from this card.

Read/Write Memory

The read/write memory is used for temporary storage of information. The memory system is implemented with 2102A-4 random access memories. These static memories are organized as 1K (1K=1024) by 1 bits. The minimum configuration is therefore 8 IC's to provide 8K bytes. In this application two cards of 2K bytes each are used.

As with the program memory card, the lower 10 bits of the address are used to select memory locations in the individual chips. The address lines are buffered by 7404 inverting buffers. Since the read/write memory is to occupy the memory space immediately after the program memory, the addresses allocated are 2000 to 23FF for card #1, and 2400 to 27FF for card #2. A circuit diagram of the cards is given in figure 15.

By referring to the memory map (figure 4), it can be seen that addresses A10 through A15 are used to select this portion of memory. Each card is selected whenever A15, A14, A12, and A11 are low and A13 is high. A10 is used to select between the two cards. Since the individual memories must be able to read as well as write data to the bus, the read/write (R/W) input on the chips is connected to the write (WR) signal. As with the program memory card, the selection is conditioned with the IO/M signal to ensure that no conflicts arise when performing input/output operations.

As well as the two cards described above, a third read/write memory card is utilized. This card is a commercially available 4K by 8 bit memory board from Kathryn Atwood Enterprises. This card is used to provide the memory for the cassette buffer.

Keyboard/Printer Control Card

The keyboard/printer control card contains the interface to the user. The keyboard is used to input commands and/or data to the computer, while the printer is used to output data to the user. A general purpose input port is also included which is used to read the various switches on the front panel. A circuit diagram of this card is given in figure 16.

The ports on this card are configured to occupy addresses 00 to 07. An on-board latch, which is identical to the one on the CPU card, is used to latch the address. The reason for duplicating the address latch is to conserve connections on the backplane. The top five bits of the inverted address bus (A3 to A7) are used to generate a board select signal. This signal is then used to enable the decoding logic as well as the data bus buffers. The lower 3 addresses are used as inputs to a 3 to 8 decoder (74LS138), the outputs of which are used to select the various ports on the card. The decoding circuitry is also enabled with the IO signal to avoid conflicts with memory addresses.

Enable signals to the data bus buffers are similiar to those on the read/write memory card with the exception that, in this case, the buffers are enabled with the IO (input/output) signal rather than the M (memory) signal. The data bus buffers used on this card are 8T26A bi-directional bus buffers.

Keyboard Control

The output from the keyboard consists of a 7 bit ASCII (American Standard Code for Information Interchange) word corresponding to the key depressed, plus a strobe signal. These 8 bits are read by the keyboard input port. Whenever the strobe signal is high, the 7 bit ASCII word is valid. The program therefore monitors the strobe input, and when it goes high, the port is read. The keyboard port is enabled by the Y0 output of the 74LS138, therefore the address is 07. The port select is also conditioned with the RD (read) signal since the keyboard port is an input port.

Printer Control

The printer used is a Bowmar model TP-3120 thermal printer. The printhead consists of a 5 by 5 matrix of heating elements. When one of these heating elements is turned on, the specially treated paper changes color at that location. In order to print a character, the ASCII data are presented to the character generator in the printer. The character generator decodes the ASCII data and enables the correct heating elements in the print matrix. The application of a print strobe turns on the elements which have been enabled, thus printing the character. In order to move the printhead, the stepper motor must be rotated. Upon receipt of four stepper motor control signals, the internal circuitry turns the motor to the requested position. The printer control section of the card consists of two output ports and a programmable interval timer. One port is used to output the ASCII data to the character generator in the printer. The second port is used to output the stepper motor signals to the printer. The interval timer is used to generate the print strobe and to time the various phases of the motor control.

The ASCII output port is enabled by output Y2 of the 74LS138, therefore it is located at address 05. Since it is an output port, it is also enabled with the write (WR) signal. The outputs from the port are buffered with 7406 open collector buffers. These buffers act as level shifters to change the logic levels of the output port to that of the character generator. The pull-up resistors associated with the open collector outputs are provided in the printer.

The two highest bits of the ASCII word are enabled with the print strobe. The reason for enabling these bits with the print strobe, is to protect the thermal printhead on the printer. Whenever an invalid character is presented to the character generator, the printhead is turned off. When there is no print strobe present, the two most significant bits of the word to the character generator are zero. Since there is no valid character code which has the two most significant bits zero, the printhead is disabled. It has therefore been ensured that the printhead is turned on only when there

7

is a print strobe present.

The output to the stepper motor is very similiar to the ASCII output. In this case the port address is 04. The open collector buffers in this case are pulled up to 15 volts with 1.5K resistors, which produces the correct logic levels for the stepper motor controls. By outputting the appropriate control code to this port, it is possible to rotate the stepper motor and hence move the printhead.

The programmable interval timer is selected with the output of an 8 input NAND gate. The address inputs to the NAND gate result in the timer occupying locations 00 to 03. The 8253 interval timer contains three independent 16 bit timers. Two of these timers are used by the printer control logic. Timer 0 is used in the one-shot mode to generate the print strobe. Whenever this timer is triggered by a bit of the stepper motor port, it provides a 13.5 msec output pulse. This pulse is used to turn on the printhead and print the character. The second timer is used to time the stepper motor control signals. The control signals must be applied for 20 msec to allow the motor time to respond. The output from this timer is used to set a 74112 J-K flip flop. The output of this flip flop is read by the computer to see if the particular time interval is complete. In the case of the interrupt driven printer routines, this output is used to interrupt the CPU via

RST 5.5. When the stepper motor control signals are turned off (00 sent to the port), the flip flop is cleared.

The time intervals for the printer control could easily have been implemented in software. A software implementation, however, is not feasible in an interrupt driven system. Since the printer routine is continually interrupted, the time delays would be stretched by an amount depending on the number of times the delay routine is interrupted. By using a hardware timer, the interrupt programs do not affect the length of the time delay.

The 7492 divider provides the clock signal to the interval timer. The divider is necessary because the timer has a maximum input frequency of 2 MHz while the system clock frequency is 3 MHz. A divide by twelve circuit in the input results in a 250 KHz input signal, and a 4 microsecond base interval.

The third timer is used by the scanning routines to generate the time delay between successive antenna readings. In this case the output is directly read by the computer to see when the time interval is over.

General Purpose Input Port

The general purpose input port occupies port address 06. This port allows the computer to read the status of various switches and signals. The outputs of the interval timer are read by bits 6 and 7 of this port. Bit 0 reads the status of the end of travel (EOT) switch on the printer. This switch is closed whenever the printhead has reached the end of the carriage. The mode control switch is read by bit 1 of the port, while bit 2 reads the status of the alarm flip flop (see cassette control card). The last input on this port is the end of conversion (EOC) signal from the analog to digital converter on the antenna/radio control card. This signal indicates whether or not the A/D converter has finished reading the input signal.

Cassette Interface Card

The cassette interface card provides the control logic for the cassette tape system. Data for the cassette tape are stored in a buffer memory until the buffer becomes full. The cassette control logic then turns on the cassette drive and writes the data onto the tape. The cassette drive is then turned off. When the cassette tape becomes full, an alarm is sounded.

The ports on the cassette interface card occupy addresses 08 to 0A. The data bus buffers for this card are enabled similiarly to those on the keyboard/printer control card. In this case however, the address decoding is such that the buffers are enabled for addresses in the range 08 to 0F.

The cassette control port is a standard output port located at address 0A. Three of the outputs on this port are used. Bit 0 is used to turn on the cassette drive. When the output goes high, the output of the open collector buffer goes low. A reed relay coil, connected between the output and +5 volts, is energized and the contacts close. The 6 volt power supply is then connected to the cassette drive. Bit 1 of this port is used to toggle the alarm flip flop. The alarm is discussed further in the section on the cassette alarm. Bit 2 is used to turn on the CASSETTE DONE LED. This LED (Light Emitting Diode) is provided on the front panel to give a visual indication that the cassette tape is ready for removal. This output is buffered with an open collector buffer which directly drives the LED.

Also included on this card is the selection logic for the 4K read/write memory card. This card provides the buffer memory for the cassette tape system. The top four addresses are decoded in such a way that the memory is located directly after the two read/write memory cards discussed earlier.

Data Output

The parallel to serial conversion of the cassette data is performed with an 8251 Programmable Communication Interface. A parallel data word is written to an input port of the 8251, where it is converted to serial format, and output along with the associated start, stop, and parity bits. When the 8251 is ready to receive more data , it interrupts the CPU via the RST 7.5 interrupt input.

The clock input to the communications interface is a 1.5 MHz signal derived from the 3 MHz system clock with a 7473 J-K flip flop. The divide by two circuit is necessary since the maximum clock frequency of the 8251 is 2.4 MHz. The baud rate line from the central processor card (2400 Hz) is connected to the transmit data clock input (TxC) of the 8251. This signal sets the rate at which data bits are sent to the cassette drive.

The 8251 occupies two addresses, but since the read (RD) and write (WR) signals are also connected to the chip, the internal hardware contains four distinct registers, two are read only and two are write only. The chip select input consists of the board select addresses, the A1 address line, and the IO line. Since the chip is selected when A1 is low, the port addresses of the 8251 are 08 and 09. The A0 address line is connected to the control/data line (C/D). This bit is used to select one of the two internal register pairs.

The resulting addresses for the 8251 ports are as follows:

08 - data input/output

09 - control word input/output

The data output (TxD) from the 8251 is fed into a circuit which is used to convert the NRZ (non return to zero) data from the 8251 into phase encoded data to be output to the cassette drive. The output of this conversion circuitry drives a 741 op-amp which is used as a comparator. One input to the op-amp is set to a level of 1.4 volts. When the output of the conversion circuitry is high, the output of the 741 goes low. When the input is at a logic 0 level, the output goes high. The high and low output levels of the op-amp are set by the zener diodes in the feedback loop. The configuration shown in figure 17 results in an output swing from -10 to +10 volts. The output of the op-amp is used as a current source to supply current to the recording head. Since the series resistor (10K) is very large compared to the head resistance, the output does indeed act as a current source. The value of the series resistor was chosen by selecting a value which would supply about 125% of the current necessary to saturate the tape. This value was selected by trying various values, and measuring the signal strengh when the tape was read back.

The cassette recorder used is a modified audio recorder, Radio Shack model CTR-43. The internal connections to the recording head were replaced with a line to the head driver on the cassette control card.

Cassette Alarm

The alarm is used to indicate that the cassette tape is almost full and should be changed. The alarm itself is simply an oscillator driving a speaker. The frequency used is 600 Hz. This frequency was obtained by dividing down the baud rate (2400 Hz) with a 7473 J-K flip flop. The output is enabled with the output of the alarm flip flop (see figure 17).

The alarm flip flop is set by toggling one bit of the cassette control port. The output of the flip flop then enables the alarm, which is then sounded. The alarm flip flop can be reset with either the CASSETTE RESET or the master RESET. The ouput of the flip flop can also be read by the CPU to determine whether or not the alarm is on.

Converting NRZ to Phase Encoded Data

The non return to zero (NRZ) data format is one of the simplest data formats. The logic level is directly proportional to the data value, i.e. a 1 is represented as a high signal and a zero is represented by a low signal. Phase encoded data are slightly more complicated. In this format the data are stored in the transitions, i.e. a high to low transition represents a

0 while a low to high transition represents a 1. Only the transitions which occur at the midpoint of a data cell are considered. The two data formats can be compared by studying figure 9.

Although the NRZ data format is simpler, there are several advantages to using the phase encoded format. The most important factor favoring the phase encoded method is the signal bandwidth. From figure 9 it can be seen that phase encoded data have a minimum of one and a maximum of two transitions per data cell. This format, therefore, results in a very narrow signal bandwidth. If a long string of zeros or ones are encoded using the NRZ format, the frequency can be seen to extend down to almost DC. The result is then a very large bandwidth when using the NRZ data format (extending from almost DC up to the data transfer rate). The chief advantage of a narrow signal bandwidth is that it allows a higher signal to noise ratio. The low frequency response required by the NRZ format is particulary harmful in this application because of the low frequency noise components introduced by the cassette drive.

The second major advantage of the phase encoded format is that it is self clocking. Since there is at least 1 transition per data cell, the data transfer clock rate can be easily derived from the data itself. The result is that the tape transport does not need elaborate speed control since the system that reads the

CLOCK	OUTPUT
1	Input
0	$\overline{\text{input}}$

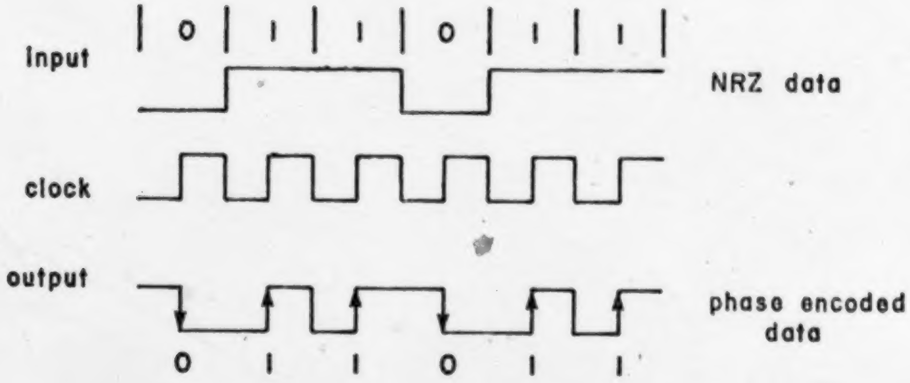

0
1
1
0
1
1
input
NRZ data
clock
output
phase encoded data
0
1
1
0
1
1

Fig. 9 Converting NRZ to Phase Encoded Data

tape can use the data to generate the clock rather than relying on its own hardware clock. The truth table in figure 9 illustrates the algorithm used to convert the NRZ data into the phase encoded format. The circuit contained in figure 17 is the implementation of this algorithm.

Antenna/Radio Control Card

The antenna/radio control card controls the scanning of the antenna system. The procedure for scanning the antenna system is to first connect a particular antenna to the radio by energizing the appropriate relay. All of the active channels are then monitored by changing the channel on the CB radio. The next antenna is then connected and the channels are monitored again. This procedure continues until the entire antenna system has been scanned. The output signal from the radio is converted to a digital value by an analog to digital converter.

The input/output ports on this card occupy port addresses 10 to 15. The data bus buffers are enabled similiarly to those on the keyboard/printer control card. The address decoding on this card is such that the buffers are enabled for addresses in the range 10 to 17. The circuit diagram for this card is given in figure 18.

Antenna Control

Most of the logic used to select the appropriate antenna is contained on the relay control card. The antenna/radio control card has only to supply a binary number corresponding to the antenna to be selected, and a signal indicating whether the X or Y antennas are to be selected. The port at address 10 is used to output this code to the relay control box.

Radio Control

Port 14 is used to output the channel code to the radio. The outputs are buffered with open collector buffers which drive the optical isolators in the radio. The procedure for selecting a particular radio channel is simply to output the appropriate code to port 14.

When the analog signal is returned from the radio it must be conditioned before it is read by the A/D converter. By referring to figure 18 it can be seen that the input is first buffered with a 747 op-amp. The second stage is basically a gain stage with an offset adjustment. The gain is necessary to increase the low signal level obtained from the radio. The meter has a full scale reading of 10 volts, while the A/D converter has a full scale reading of 5 volts. In order to accomodate these two devices the signal is first amplified to a maximum of 10 volts and sent to the meter. The signal is then scaled by two before it is presented to the A/D converter.

The offset adjustment on the second stage is very important. The output signal from the radio can vary between positive and negative signals. The offset adjustment shifts the signal so that the output is always positive. At this point the signal is sent to the meter. The reason for placing the meter at this point is so that any noise picked up in the long cable to the meter can be filtered out in the next section.

The filter section is a simple one stage RC filter. A FET is placed in parallel with the resistor so that the filter response can be changed. The ability to change the filter response is very important when switching channels. In order to filter out low frequency noise components, the cutoff frequency of the filter must be fairly low. As a result of the low cutoff frequency, the filter cannot respond to sudden changes in the signal level, as happens when the channel is changed. In order to allow the filter to respond to channel changes, the filter bandwidth must be increased when the channel is changing. The bandwidth is increased by turning on the FET and effectively reducing the resistance in the RC filter.

The filter bandwidth is increased for 2 msec when the channel is changed. The timing for the filter change is provided by a 74123 one-shot. When the one-shot is triggered, the output goes high for 2 msec. The output then triggers a comparator constructed with a 747 op-amp. The output of the comparator turns on the FET and increases the bandwidth of the filter. The trigger signal for the one-shot is provided by the port select logic. Whenever an I/O address of 15 is detected by the 74LS138 decoder, output Y2 goes low and triggers the one-shot.

The output of the filter is buffered by another 747 op-amp, scaled by two in a resistive divider, and then presented to the analog to digital converter. The 5.1 volt zener diode in the input of the A/D converter is used to limit the input voltage to the converter.

A start convert pulse must be applied to the analog to digital converter before it will read the input signal. Whenever the port select logic detects an I/O address of 12 the 74LS138 sends a pulse to the start conversion input of the A/D converter. When the conversion is finished, the end of conversion output is set. As was discussed earlier, the end of conversion (EOC) output is monitored via the general purpose input port on the keyboard/printer control card. When the CPU receives an end of conversion signal, it reads the A/D converter through port 13. The system clock is divided by two with a 74112 J-K flip flop before it is applied to the clock input of the A/D converter. It is necessary to reduce the clock frequency since the ADC0800 has a maximum clock frequency of 2 MHz.

Power Supplies

Positive Supplies

The five volt power supply is used mainly to provide power for the logic circuitry of the computer. The fifteen volt supply is used mainly to supply the printer. The basic configuration for both supplies, shown in figure 19, is the same. A diode bridge and filter capacitor supplies the unregulated DC to the regulator circuit. A standard three terminal regulator is used to control the output voltage. In order to supply the large currents required (up to 5 amps), the regulator is paralleled with a power darlington transistor. Resistors R1 and R2 act as current dividers to determine the ratio of current through the regulator and pass transistor. The ratio is set so that, at full load, approximately 90% of the current flows through the pass transistor, and only 10% through the regulator. Resistor R3 determines the current at which the pass transistor is turned on. Capacitors C1 to C4 are used to decouple the input and output of the regulator.

A crowbar circuit is used to protect the system from overvoltage. Whenever an overvoltage condition is detected, the crowbar circuit places a short across the input of the regulator. The current drawn through the short will blow a fuse. A Motorola MC3423 chip is used to provide the crowbar action. If the output voltage climbs to a value determined by resistors R4 and R5, the output goes low. The output then draws current from the gate of the SCR and turns it on. The current through the SCR is set, by RL, to be large enough to blow the fuse. For both supplies, R4 and R5 are set so that the crowbar circuit is tripped whenever the output rises 2 volts above its nominal value.

Negative Supplies

The negative power supplies are used mainly to supply bias voltages to various devices in the system such as the printer, A/D converter, etc. The circuit is a standard full wave rectifier followed by a filter capacitor and a three terminal regulator. The regulators used are rated for 1 amp of current.

IV. SOFTWARE DESCRIPTION

The control program for the data acquisition system consists of four major sections, the main program and three interrupt service subroutines. The main program controls the interface to the user. Whenever the mode switch is set to CONTROL, the program monitors the keyboard and accepts commands. A command entered via the keyboard causes the program to jump to the appropriate keyboard service routine.

When the mode switch is set to SCAN, the system is controlled by the interrupt system. In this mode, the function of the main program is to monitor the mode control switch to detect when the user wishes to return to control mode. The most important interrupt service subroutine is the real time clock routine. Once per second the CPU is interrupted by this routine. The interrupt service subroutine increments the clock and executes the scanning procedure. The data collected from the antenna system are then processed to determine all of the locations and activity states. If it is time to output data to the printer or cassette drive, the appropriate routines are called.

When the real time clock routine decides that the data should be printed, the data are loaded into the print buffer, and the interrupt driven printer routine is called. This routine enables the interrupt system so that subsequent calls to the routine are controlled by the interrupt system. This routine loads a character into the printer, and sets the timer for the print strobe. Control is then returned to the interrupted program. Upon termination of the print strobe, an interrupt is generated so that the program can move the printhead to the next print position. When the stepper motor control signals have been applied, control again returns to the interrupted program. The next interrupt indicates that the printer is ready to print the next character.

The real time clock routine can also enable the cassette interface interrupt routine. Whenever the cassette system is ready to output another character to the cassette drive, it interrupts the system. The interrupt service subroutine then outputs the next character to the cassette system.

An interrupt driven system is used because several things must happen at once. The system must be able to scan the antenna system, print the data, and output data to the cassette drive all at the same time. The only feasible way to allow all of these operations to occur at once, is to use an interrupt driven system.

Various section~~s~~ of the control program are now discussed.

Initialization

The initialization program is executed whenever the system is reset. Various parameters in the system, such as the scan rate, can be set by the user. When the system is initialized, these parameters are set to their default values. The default values are therefore the values of these parameters whenever the system has been reset. If these default values are not suitable, they can be changed by the user. The initialization routine must also reset the real time clock and initialize the printer. When all of these tasks have been completed, the reset message is printed and the interrupt system is enabled. Control then passes to the main program.

Main Program

The actual main progam is only a few statements long. The status of the mode switch is first checked. If the mode switch is set to CONTROL, the keyboard program is executed. If the mode switch is set to SCAN the scan flag is set. This flag indicates to the real time clock interrupt routine that the antenna system should be scanned after the clock is incremented.

Keyboard Routines

The keyboard routine (KEYB) accepts characters from the keyboard and decides if the input corresponds to a valid command. If it does, the correct keyboard service routine is executed. If the input is not a valid command, an error message is printed.

All commands consist of three characters. The keyboard routine reads these characters from the keyboard and stores them in a buffer. When three characters have been received, the program compares the three character code to a list of valid codes stored in a table (KYTAB). This table contains the address of the keyboard service routine as well as the character code. When a match is found, the address of the keyboard service subroutine is read from the table and the service routine is executed.

The subroutine CHKEY is called by the keyboard routine. CHKEY waits for a character from the keyboard, prints the character, and returns the ASCII character in the accumulator. If no character is received for two seconds, the printhead is moved two positions to the right so that the last character printed can be seen. This routine also converts any lower case letters into upper case letters. This conversion is necessary since the printer can only print upper case letters.

System Monitor

The system monitor consists of five keyboard service subroutines which can be used to study or control the system.

The DPM command is used to display memory contents. The starting and ending memory addresses are received from the keyboard. The program then prints the data in the memory area specified.

In order to set memory locations, the STM routine is called. In this routine, the address of the memory location to be changed is received from the keyboard. The program then accepts the data from the keyboard and stores it at the specified memory location. The address is then incremented and printed. A new data byte is then received from the keyboard and stored in memory. This process continues until a period is received instead of a data byte. Control is then returned to the keyboard program. This routine is useful for loading a program into read/write memory. The RUN command can then be used to execute the program.

The RUN program is used to manually execute programs. This routine receives an address from the keyboard. Control is then transferred to the program residing at the memory location specified.

Keyboard service routines OUTP and INP are used to control data flow to and from the various ports in the system. These routines must first store a program in read/write memory and then call the program.

Display Routines

The display routines allow the user to check the values of various parameters in the system. Once a number of parameters have been changed, it is sometimes useful to recheck all the values to see that they have been set properly. Besides allowing the user to check specific parameters, the routine can also print any parameters which have been changed from their default values.

DSP is the keyboard service routine which controls the display function. Following the command, a space is received. The space is followed by the three character code for the parameter to be displayed. The routine to check the character code and jump to the appropriate display service subroutine is similiar to the keyboard routine (KEYB). The only difference is the fact that DSP compares the code to a set of valid codes in the display table (DISTB).

Two utility routines are included in this section to facilitate printing data. PHL is a printer routine which prints the hexadecimal data in the HL register. This routine is useful for printing hexadecimal data such as addresses. In order to print decimal data, the subroutine PRBYT is provided. PRBYT takes the byte contained in the accumulator, converts it to decimal, and prints the number. This routine is used to print most of the parameters.

Corresponding to each possible parameter is a display service routine. Whenever the code for a particular parameter is received by the DSP routine, the appropriate display service routine is executed. Most of the routines are fairly simple. The parameter is fetched from memory, converted to printable form, and then printed.

The display routine corresponding to the three character code 'ALL' prints all of the parameters which have been changed from their default values. The procedure for accomplishing this task is to successively compare each possible parameter with its default value. If the parameter is still at its default setting, the next parameter is checked. If the parameter value has been changed, the new value is printed before the next parameter is checked. The program continues in this manner until all parameters have been checked.

Utility Subroutines

The utility subroutines are a collection of useful subroutines which are utilized by many portions of the program.

One utility program is the multiplication subroutine (MULT8). This subroutine performs an 8 bit by 8 bit multiplication on the numbers contained in the B and C registers. The 16 bit result is returned in the BC register. A standard shift and add algorithm is employed to perform the multiplication. As with all of the utility programs, the only registers altered are those required to pass parameters out of the routine.

Two routines (DECHX and HXDEC) are used to convert decimal numbers to hexadecimal, or hexadecimal numbers to decimal. When converting a hexadecimal number to decimal, there is a possibility of an error since the decimal number is restricted to two digits. For this reason, the carry bit is set if the conversion cannot be performed. The calling program should therefore check the carry bit before accepting the output of this routine.

In order to convert ASCII characters to hexadecimal and vice versa, two conversion routines CONHX and CONAS are used. In these routines, the carry bit is set if the conversion cannot be performed. The main purpose of these routines is to facilitate interfacing to the printer and the keyboard.

In order to set many of the parameters, a number is entered via the keyboard. Since it is desireable to be able to enter the number in decimal, it is necessary to convert the two decimal characters to a hexadecimal number. The subroutine DECKY receives two characters from the keyboard and produces a hexadecimal result.

Two suboutines are also provided to handle 16 bit numbers. HLSTK takes the hexadecimal number contained in the HL register and loads the four corresponding ASCII characters into the first four positions of the print stack. GTADD receives four ASCII characters from the keyboard, converts them to a 16 bit binary number, and returns the result in the HL register.

A number of routines are also provided to change the various register pairs by fixed amounts. AD882 adds 882 to the HL register, while AD21 adds 21. INDE is used to increment the DE register by 21.

Also included in this section is the ERROR routine which prints the error message on the printer.

Stepper Motor Control Programs

The procedure for moving the printhead on the printer is to send the appropriate control signals for the new position to the stepper motor for 20 msec. The new position can be one position away, either to the right or the left. In order to keep track of the current printhead position, a table of stepper motor control signals (STEP) is used. A variable (PRPOS) points to the location in the table that contains the stepper motor control signals for the current position. In order to move the printhead to the right or left, the stepper motor control signal before or after the current one is read. This new control signal is then output to the printer. The two routines provided (MOVL and MOVR), move the printhead one position to the left or to the right of its current position.

A pointer is also used by the printing routines to point to the position in the print stack which corresponds to the current printhead position. Whenever the printhead is moved, this pointer is adjusted so that it points to the proper location in the print stack.

A routine (HOME) is also provided to move the printhead to the home position. This routine keeps calling the MOVR subroutine until the end of travel (EOT) switch is closed. The EOT switch is closed when the printhead reaches the end of the carriage. The home position is two positions to the left of this point, therefore the MOVL subroutine is then called twice.

The RETN subroutine is used to move the printhead from the home position to the first print position on the left side of the paper. This return is accomplished by calling the MOVL subroutine 18 times.

Printer Control Programs

The printer control programs control the printing of characters on the printer. The procedure for printing a character is to output the ASCII character to the printer, and then apply the print strobe for 13.5 msec. In order to allow the printhead time to cool down, the printhead should not be moved for at least 6 msec.

The print routine (PRINT) uses timer number 0 to generate the print strobe. After the ASCII character has been sent to the printer, the timer is initialized. First the flip flop connected to the output of the timer is enabled. The correct timer count is then loaded into the timer and the gate of the timer is raised. The timer then outputs the print strobe. The program then monitors the timer flip flop until the print period is over. The flip flop is then cleared and re-enabled. A new timer count is sent to the timer to set a delay of 6 msec. Again the flip flop is monitored until the time delay is over. The printhead is then moved one position to the right in preparation for printing the next character.

In order to print the entire contents of the print stack, the subroutine PSTAK is called. This routine moves the printhead from the home position to the first print position, and then calls PRINT 18 times. This routine prints all of the characters in the stack, and then returns the printhead to the home position.

A convenient way to print messages on the printer is via the PRIMM subroutine. This routine prints any ASCII characters that it finds directly after the CALL instruction. It continues printing characters until a null (00) character is encountered in the list. The routine automatically adjusts the return address on the stack so that program execution continues with the instruction immediately following the null character.

A subroutine (CLPSK) is also provided to clear the print stack. This routine is necessary if a short line is printed via the PSTAK subroutine immediately after a longer line. Since the end of the longer line is still in the print stack when the shorter line is printed, the end portion of the longer line is printed again.

In order to print text on the printer, the keyboard service subroutine PRT is provided. The routine first generates a carriage return and a line feed before accepting text. Characters are then printed as they are entered from the keyboard. Special control characters are provided to allow the user to generate a carriage return and line feed, or a backspace. A control character is also provided to indicate that no more text will be provided. Upon receipt of this contol character, the program returns control to the monitor.

Antenna Control Programs

ANT is a keyboard service routine which allows the user to close a particular antenna relay. An X or Y is first received from the keyboard to indicate which set of antennas is desired. The antenna number is then received from the keyboard, converted to binary, and then output to the antenna control port along with the control bit $(X/\overline{Y})$, which specifies which set of antennas is desired.

After the antenna relay has been activated, the system reads the A/D converter. This function is not necessary, but is provided to facilitate troubleshooting the system. In order to convert the A/D value to a voltage, the binary input is multiplied by a constant to produce a 16 bit number. The constant was chosen so that the most significant byte of the result corresponds to the voltage (measured in tenths of a volt). This voltage is then printed.

Hardware Timer Delay

This subroutine (DEL20) is called by the stepper motor control programs. This routine outputs the control signals for the stepper motor and then delays for 20 msec. Timer number 1 is used to time the delay. After the count is loaded into the timer, the program monitors the timer flip flop until the output indicates that the delay period is over.

CB Radio Control Programs

Two keyboard service routines are included in this section. The SAC program allows the user to set the channels which are to be monitored. This routine also initializes the temporary print buffer in preparation for loading data into the buffer. The CHN routine allows the user to manually set the radio channel from the keyboard.

Both SAC and CHN call a subroutine (GCHAN) which receives a channel number from the keyboard, and fetches the corresponding channel code from a table. This routine returns the ASCII channel number, the hexadecimal channel number, and the channel code.

The SAC routine receives a series of channel codes via the GCHAN subroutine. These channel codes are placed in a table (CHN1 to CHN5) where they are used by the scanning programs. Any locations in the table not filled with channel codes are filled with null characters. A variable (CNUM) is set to the number of channels received. The ASCII characters for the channel numbers are stored in the temporary buffer. (see section 'Real Time Clock Interrupt Routine')

A variable NDAT is also set. This variable is used by the cassette routine to calculate when the buffer is full. Since it takes approximately two seconds to initialize the cassette system, the initialization should be started two seconds before the buffer is full. NDAT is the maximum number of characters which will be placed in the buffer within the next two seconds. NDAT is the product of the number of characters per line (22), the number of seconds (2), and the number of channels used (CNUM).

CHN is a keyboard service routine which allows the user to select the radio channel. This routine calls the GCHAN routine which returns the channel code and the channel number. The channel code is output to the radio. The channel number is stored in CHNL where it can be read by the display routine.

Interrupt Driven Printer Routine

The interrupt driven printer routine is called when data must be printed at the same time as other routines are executing. This routine outputs appropriate commands to the printer control card, sets the timer and then exits from the interrupt routine. The routine is not reentered until the timer indicates that the delay is over by interrupting the system.

When the interrupt system returns control to the program, execution must begin at the same place that execution stopped at the end of the previous call to the routine. A pointer (LOC) contains the address of the next instruction to be executed. When the interrupt routine is entered, this address is loaded into the program counter so that execution begins at the proper location. Upon completion of the interrupt routine, the address corresponding to the next instruction to be performed is loaded into LOC. Upon receipt of the next interrupt signal, execution continues from this point.

Initially the interrupt for this program is disabled. In order to start execution, the interrupt routine (PTINT) is called like a subroutine. The program then enables the interrupt and places the program under control of the interrupt system. When the interrupt routine has completed its task, the interrupt is disabled and the location pointer (LOC) is set to the start of the routine.

Characters to be printed are stored in a buffer along with the carriage control characters. A variable (PRB) points to the next character in the buffer to be printed. The characters are loaded from the print buffer and printed until a carriage control character is encountered. Upon receipt of a carriage return character, the program executes a carriage return and a line feed. Printing then continues with the next character in the print buffer. If an end of file character is encountered, the interrupt system is disabled and control returns to the interrupted program.

Interrupt Driven Printer Subroutines

Four keyboard service routines are provided to control the printer. The PRI routine enables the printer and sets the print interval. POF disables the printer. A flag (PROF) is used to indicate whether the printer is enabled or not. This flag is read by the scanning routine to decide whether to print the data or not.

Ordinarily the transmitter positions are printed by the scan routine. A flag (ACLOC) is used to indicate what data should be printed. When ACLOC is set, the flags are printed instead of the positions. When ACLOC is reset, the positions are printed. The POS routine resets the flag, while FLG sets it.

Real Time Clock Subroutines

Two keyboard service routines are provided to set and read the real time clock. TST is used to set the clock. The interrupt system is first disabled so that the clock is not incremented while the program is trying to set it. The hour and minute information is received from the keyboard and stored in the clock counters. Upon receipt of a space, the interrupt system is enabled and the clock starts.

The TIM routine is used to print the time. The interrupt routine is first disabled and the time stored in a temporary location. The interrupt system is then reenabled and the time is output to the printer.

Scan Subroutines

This section contains a number of routines which read the keyboard and set the appropriate internal parameters. Parameters which can be set include the scan interval, the cassette store interval, the multiplier constant for the interpolation algorithm, the test antenna comparator value, the distance at which the scan rate will be increased, and the percentage for the activity algorithm.

Two keyboard service routines are also included to control the population matrix facility. PPM prints the population matrix for one transmitter. A number is received from the keyboard which indicates the transmitter data to be printed. The program then fetches the data for the correct transmitter and outputs it to the printer. CLR clears the entire population matrix.

Two utility routines are included to facilitate printing the population data (see section 'Real Time Clock Interrupt Routine'). PRT16 takes a 16 bit binary number, converts it to decimal, and prints the result. The routine is written in such a manner that the full range of 0 to 65,536 can be printed. Individual digits are printed with the PRTNB routine.

Interrupt Routine for Cassette Interface

Whenever the cassette system is ready to receive another character, the CPU is interrupted via RST 7.5. The cassette interrupt service subroutine then outputs a character to the cassette interface port.

The routine first checks CRDY to see if the cassette drive is ready to receive data. CRDY is cleared by the scanning program if the cassette drive has not reached proper operating speed. If the cassette drive is not ready to receive data, a delete character is output. If the cassette system is ready, a character is loaded from the cassette buffer and sent to the communication interface. The cassette buffer is then checked to see if there are more characters to be output. If there are more characters, control is returned to the interrupted program. The next character is output upon receipt of the next interrupt.

When the cassette buffer is empty, the program checks to see if the cassette tape is full. The tape is checked by reading the CFIN flag, which is set if the user has reset the cassette system after inserting a new tape. If CFIN is set, the cassette system is reset and the CASSETTE DONE lamp is turned on before the cassette drive is turned off.

Before returning control to the interrupted program, the interrupt flip flop is reset to reenable the interrupt system.

Cassette Interface Subroutines

Data in the cassette buffer are stored in the following format:

<GS>ASCII data<DEL><CR>

The group separator <GS> is used to indicate to the cassette tape reader that a new line is starting. The delete character <DEL> is inserted in each line to synchronize the receiving system if it has slipped out of phase. A delete character is used since its ASCII representation consists of a string of ones. The carriage return <CR> is used to indicate the end of a line. The receiving system inserts a line feed as well as a carriage return.

In order to facilitate loading control characters into the cassette buffer, a series of utility routines are provided. CRLF loads a delete and a carriage return into the buffer. NLINE loads a group separator. LEOF is a subroutine which loads a delete plus two file separators. A one second delay routine is also included.

The last line on the cassette tape is stored in the following format:

<GS>ASCII data<DEL><CR><DEL><FS><FS>

In this case two file separators are added at the end of the line. The receiving system, upon receipt of the file separators, stops reading data from the tape.

The cassette buffer is a section of read/write memory reserved for this purpose. In order to change the location of the cassette buffer, the CST and CEN subroutines are used to set the starting and ending addresses of the buffer.

A keyboard service routine (ALM) is also included which turns on the cassette alarm. The routine is included strictly for maintenance purposes.

If the user wishes to control the cassette drive from the keyboard, the two routines CON and COF are used. CON turns on the cassette drive, while COF turns it off. The cassette timer is used to keep track of the amount of tape left. The CTP routine receives the tape length from the keyboard, converts the time to seconds, and loads the cassette timer with the result. Whenever the cassette drive is on, this timer is decremented every second. When it reaches zero, the tape is full.

If the operator wishes to close the cassette tape file manually, the STP command is issued. The function of this program is to output file separators to the cassette tape so that the receiving system knows that there is no more data on the tape. The cassette drive is turned on and the cassette system initialized. Two file separators are then loaded into the cassette buffer directly after the last data line. The cassette pointers are then set so that the entire buffer will be output. After a delay of one second, to allow the cassette drive to reach operating speed, the CRDY flag is set. The CRDY flag indicates to the cassette interface service subroutine that the data can be output.

A keyboard service routine is also provided which allows the user to type characters directly into the cassette buffer. This routine is essentially the same as the PET routine discussed earlier. In this case the characters are also loaded into the cassette buffer. The control characters required by the cassette system are also loaded before and after each line.

Real Time Clock Interrupt Routine

As data are collected from the antenna system, they are processed and stored in a temporary buffer. If the data are to be printed, they are moved from the temporary buffer into the print buffer and then printed. The reason for using two buffers is because the printer takes several seconds to print the entire contents of the buffer. Since the data may change every second, there is a possibility that the data will be changing as the printer is trying to print it. The data are therefore transferred from the temporary buffer into the print buffer. Any changes are then made to the temporary buffer while the data in the print buffer are printed. The format of the temporary buffer is as follows:

```
TTTTTT CC XXX YYY ABT <CC>
TTTTTT CC XXX YYY ABT <CC>
TTTTTT CC XXX YYY ABT <CC>
TTTTTT CC XXX YYY ABT <CC>
TTTTTT CC XXX YYY ABT <CC>
```

where TTTTTT is the time (hr, min, sec)

CC is the channel number

XXX is the position on the X axis (antenna XX.X)

YYY is the position on the Y axis (antenna YY.Y)

A is the activity flag, with a 0 signifying a motionless transmitter, and a 1 indicating a transmitter in motion.

B is a flag indicating whether the transmitter is above or below the snow. This feature is not implemented.

T is the test antenna flag. A 1 indicates that the radio interference is above the predetermined level.

<CC> is the carriage control character. A carriage return (0D) indicates the end of a data line. An end of file character (1C) indicates that the current line is the last data line in the buffer.

Several sections of the buffer are set by the SAC subroutine when the active channels are set. The channel numbers are loaded into the temporary buffer as well as the carriage control characters. Spaces are also inserted between the data sets. The correct time, positions, and the flags, are inserted by the scan subroutines.

The real time clock interrupt routine is executed upon receipt of the RST 6.5 interrupt. This routine controls the scanning of the antenna system as well as incrementing the real time clock. The interrupt routine first resets the interrupt flip flop so that further interrupts can be processed. The real time clock is then incremented. A flow chart for this routine is given in figure 10.

Once the clock has been incremented, the alarm system is serviced. If the alarm is on, the alarm flip flop is tested to see if the alarm has been reset. If the alarm has

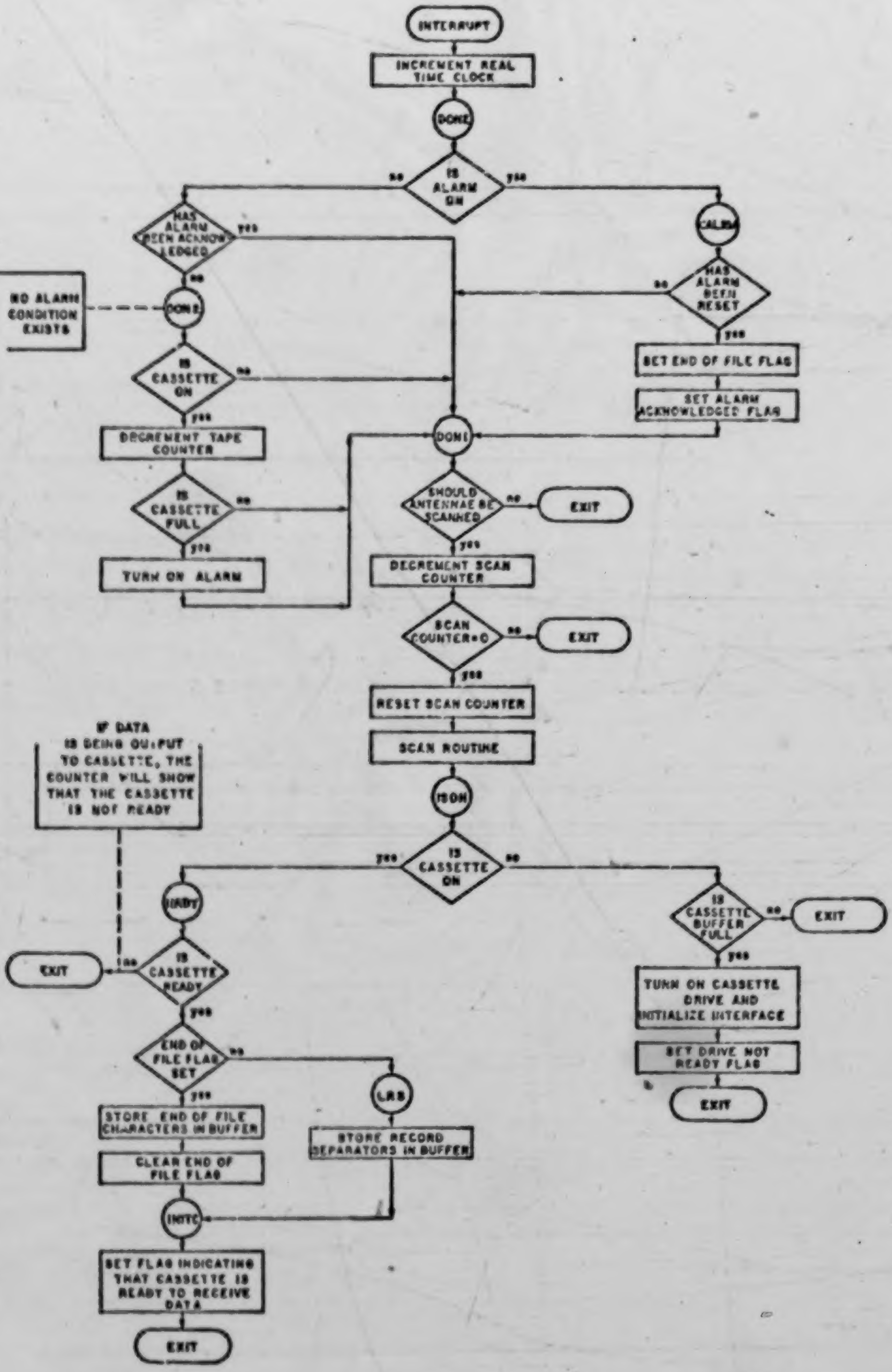

INTERRUPT
INCREMENT REAL TIME CLOCK
DONE
IS ALARM ON
no
yes
HAS ALARM BEEN ACKNOWLEDGED
yes
no
NO ALARM CONDITION EXISTS
DONE2
IS CASSETTE ON
no
yes
DECREMENT TAPE COUNTER
IS CASSETTE FULL
no
yes
TURN ON ALARM
CALRM
HAS ALARM BEEN RESET
no
yes
SET END OF FILE FLAG
SET ALARM ACKNOWLEDGED FLAG
DONE1
SHOULD ANTENNAE BE SCANNED
no
EXIT
yes
DECREMENT SCAN COUNTER
SCAN COUNTER=0
no
EXIT
yes
RESET SCAN COUNTER
SCAN ROUTINE
ISON
IS CASSETTE ON
yes
no
IF DATA IS BEING OUTPUT TO CASSETTE, THE COUNTER WILL SHOW THAT THE CASSETTE IS NOT READY
IRDY
IS CASSETTE READY
no
EXIT
yes
END OF FILE FLAG SET
no
yes
STORE END OF FILE CHARACTERS IN BUFFER
CLEAR END OF FILE FLAG
LRS
STORE RECORD SEPARATORS IN BUFFER
INITC
SET FLAG INDICATING THAT CASSETTE IS READY TO RECEIVE DATA
EXIT
IS CASSETTE BUFFER FULL
no
EXIT
yes
TURN ON CASSETTE DRIVE AND INITIALIZE INTERFACE
SET DRIVE NOT READY FLAG
EXIT

Fig. 10 Real Time Clock Interrupt Routine

been reset, the end of file flag is set. This flag ensures that end of file characters are stored on the cassette tape after the data are stored. A flag is also set to indicate that the alarm has been acknowledged. The routine then executes the scanning procedure.

If no alarm condition exists, the cassette drive is tested to see if it is on. If it is not on, the scanning procedure is executed. If the cassette drive is on, the tape timer is decremented and checked to see if the tape is full. If the tape is full, the alarm is sounded.

After the alarm has been serviced, program execution continues at DON1 (see figure 10). At this point the program determines whether or not the antennas should be scanned. The scan frequency is controlled by the scan counter. When the counter has been decremented to zero, the counter is reset and the scanning procedure is executed.

The flow chart for the scan routine is given in figure 11. The antenna system is monitored, and all readings are stored in a table. The procedure for testing the antenna system is to select an antenna, and then record the signal strengths on this antenna for all channels. This procedure could have been reversed; that is, select a channel and then scan all antennas for that particular channel. This method has the disadvantage that the antenna relays would be used more often. For example, if all five channels are monitored, the antenna relays would be selected five times as often.

Since it takes a finite time for the radio to respond

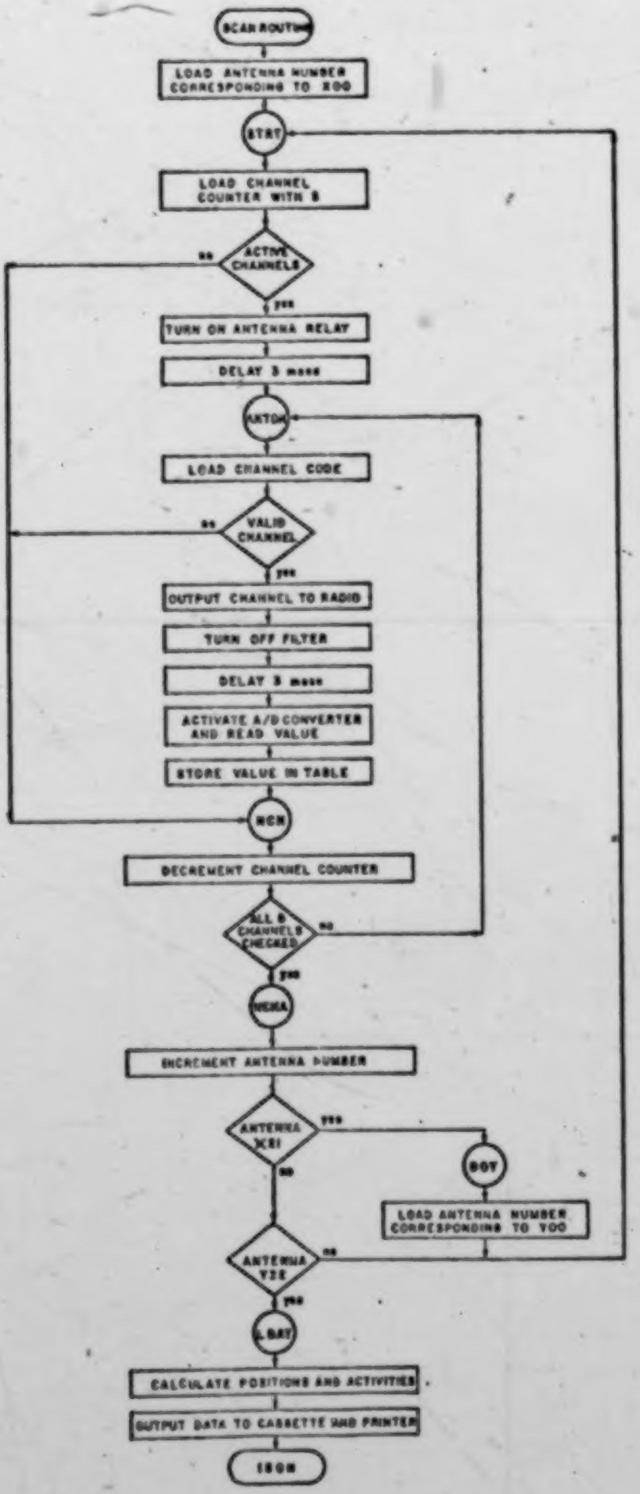

SCAN ROUTINE
LOAD ANTENNA NUMBER CORRESPONDING TO X00
STRT
LOAD CHANNEL COUNTER WITH 8
ACTIVE CHANNELS
no
yes
TURN ON ANTENNA RELAY
DELAY 3 msec
ANTCH
LOAD CHANNEL CODE
VALID CHANNEL
no
yes
OUTPUT CHANNEL TO RADIO
TURN OFF FILTER
DELAY 3 msec
ACTIVATE A/D CONVERTER AND READ VALUE
STORE VALUE IN TABLE
NCH
DECREMENT CHANNEL COUNTER
ALL 8 CHANNELS CHECKED
no
yes
NEXA
INCREMENT ANTENNA NUMBER
ANTENNA X81
yes
no
BOT
LOAD ANTENNA NUMBER CORRESPONDING TO Y00
ANTENNA Y28
no
yes
LBAT
CALCULATE POSITIONS AND ACTIVITIES
OUTPUT DATA TO CASSETTE AND PRINTER
ISON

Fig. 11 Scan Routine

to a channel change, or for the relays to close, a three millisecond delay is inserted before the A/D converter is read. The delays are implemented with the hardware timer. The hardware timer is used so that the delay time is not affected by the other interrupt routines.

After the antenna system has been scanned, the A/D readings for all channels on all antennas are stored in a table in read/write memory. The data must then be processed to determine the transmitter locations.

The subroutines used to calculate the transmitter positions, set the flags, control the cassette drive, and control the printer, are discussed in detail later. The first subroutine called (LOADA), calculates the transmitter positions based on the data in the table, and loads the data into the temporary buffer. This subroutine also increments the counters in the population matrix corresponding to the positions. The real time clock is then loaded into the temporary print buffer via the subroutine LDTME.

The activity flags are then tentatively set by the ACTAD subroutine. This subroutine calculates the activity based on the relative A/D readings. If lateral motion is detected at a later time, these flags are corrected. The algorithm for the activity states uses the scan interval as a timer. For this reason, the flags are tentatively set at this time since lateral motion is only checked every cassette store interval.

As discussed earlier, if a transmitter is found within a predetermined distance of the master transmitter, the cassette store rate is increased. The cassette store rate is controlled by the CSINC subroutine.

The scan routine now determines if the data should be output to the cassette. If the cassette store interval is not over, the scan routine is aborted. If the cassette store interval is over, the test antenna flags and the above/below flags are loaded into the temporary buffer by STTAN and STABL.

The position and activity states for all transmitters are then compared to their values at the end of the previous cassette store interval. If any of the transmitter data have changed, the data for that particular transmitter are sent to the cassette buffer. This function is implemented by the LODEC subroutine.

The print interval is then checked to see if the data should be printed. If the print interval is over, the entire temporary buffer is loaded into the print buffer by IDPRT, and the interrupt driven printer routine (PTINT) is called.

After the scan routines have been executed, the cassette system is serviced. If the cassette drive is not on, the cassette buffer is checked to see if it is full. If the buffer is full, the cassette drive is turned on and the cassette interface is initialized. A flag is also set to indicate that the drive is not ready to receive data since it has not reached operating speed yet.

If the cassette drive is on, a counter is tested to determine whether or not the drive is ready to receive data. If the drive is ready to receive data, the end of file flag is checked. If the alarm has been acknowledged (end of file flag set), end of file characters are loaded into the buffer after the data. If there has been no alarm acknowledgement, record separator characters are loaded instead. At this point a flag is set to indicate that the cassette is ready to receive data. As was seen earlier, this flag is monitored by the cassette interrupt routine to determine whether it should start transferring the buffer data to the cassette tape.

A population matrix facility is also provided. A matrix is stored in read/write memory with a counter corresponding to each antenna crossing. Each scan interval, after the positions have been determined, the counter corresponding to the correct position is incremented.

Scan Subroutines

The scan subroutines are a collection of subroutines which are called by the real time clock interrupt routine. After the data from the antennas has been stored in read/write memory, these routines process the data and calculate the transmitter positions, activities, etc.

Position Calculations

The transmitter positions are calculated by the POSAX subroutine. When calling the subroutine, the DE register is pointed at the antenna data for a particular channel. The subroutine returns the transmitter positions in ASCII, the antenna number of the antenna nearest to the transmitter, and the A/D reading on the closest antenna. Four variables are used to keep track of the various readings. APREV contains the A/D reading for the previous antenna. APRES contains the A/D reading for the antenna with the largest signal encountered so far. ALAST and ANEXT contain the readings for the antennas immediately before and immediately after the antenna with the maximum signal. The variable MAXNO contains the number of the antenna which was found to have the largest signal.

The routine first scans through the antenna data and compares each reading with the previous maximum. If the new reading is less than the previous maximum, it is ignored. If the new reading is greater than the previous maximum, the data for the new antenna (ALAST, APRES, ANEXT, and MAXNO) replace that of the previous maximum. Antenna readings are checked in this manner until all have been checked.

The routine now knows approximately where the transmitter is located along this axis. The data must now be analysed to decide if the transmitter is directly under the antenna with the largest signal, or between this antenna and one of its neighbours. The A/D reading on the maximum antenna is first multiplied by 25, and the readings on the two neigbouring antennas are multiplied by the multiplier constant (CMULT). If CMULT times a neighbouring antenna reading is greater than 25 times the maximum antenna reading, the transmitter is assumed to be between the two antennas. For example, if CMULT is set to 50, and the maximum antenna is determined to be XX, if antenna XX+1 has a signal greater than half that of antenna XX, the transmitter is assumed to be at XX.5. If both neighbouring antennas have signals sufficiently close to the maximum antenna, the transmitter is assumed to be under the maximum antenna. Before returning to the calling program, POSAX stores the ASCII characters corresponding to the antenna number in locations LCY5 to LCY5+2. The number of the nearest antenna is returned in NEARY.

The POSAX subroutine is called by the LDDT subroutine. This subroutine calculates both coordinates for a particular channel, and loads the data into the temporary buffer. The routine is entered with register A containing a number corresponding to the transmitter data to be loaded. Upon entering the routine, various pointers are set up to point at variables corresponding to the transmitter number selected. A pointer is set to the activity file. The activity file contains the A/D reading for the maximum antenna on the various channels. A pointer is also set to point to the positions in the temporary buffer where the position data are stored. The DE register is then pointed at the X antenna data for the transmitter selected, and the POSAX subroutine is called. The position returned by POSAX is then loaded into the temporary buffer. The value returned in NEARY is transferred to NEARX. POSAX is then called again to calculate the position on the Y axis. The data are again transferred into the temporary buffer. Before returning to the calling program, the A/D value on the maximum antenna is stored in the activity file.

In order to load the data for all channels into the temporary buffer, LDDAT must be called several times. LDDAT is called by the LOADA subroutine. Before loading the data into the temporary buffer, the activity file is transferred to the previous activity file. When LDDAT is called, the new A/D values are stored in the activity file. Since the activity routines compare the present A/D readings with the previous readings, a file is kept of the previous as well as the present readings. LDDAT is then called once for each active channel. The transmitter locations for all active channels are therefore loaded into the temporary buffer.

LOADA also calls LDPOP for the first three transmitters. This routine increments the counter in the population matrix corresponding to the transmitter position. Since the two nearest antenna numbers are stored in NEARX and NEARY, this routine uses these values to determine which counter should be incremented. This routine is entered with register A containing the transmitter number.

Loading the Time

The subroutine LDTME loads the real time clock into the temporary buffer. The real time clock counters are first converted to ASCII, and then loaded into all five data lines in the temporary buffer.

Activity Calculations

The scan routine also determines the activity state of the transmitters. The subroutine ACT determines the activity state for the transmitter whose number is entered in the accumulator. If no motion is detected for five scan intervals, the activity flag is set to zero. If motion is detected, the flag is set to one.

Various pointers are first set to point at various variables. The variables used are the past A/D reading, the present A/D reading, and the activity flag. The present and past A/D readings are then compared. If the two values are within a certain percentage (ACPCT) of each other, the transmitter is assumed to be still. When this condition is detected, the activity counter is decremented. If no motion is detected for five successive scan intervals, the activity flag is set to a one. If motion is detected, the activity counter is reset to five, and the activity flag is set to a zero.

In order to set the activity flags for all channels, the ACT subroutine is called five times by the ACTAD subroutine. ACTAD therefore sets the activity flags for all five channels. As discussed earlier, these flags should be overriden if lateral movement is detected.

Cassette Store Rate

The Pythagorus Theorem is used to calculate the distance between two transmitters. In order to implement this algorithm, a square root function is required. Subroutine SQRT finds the square root of the number entered in the DE register. The result is returned in the accumulator. The algorithm used is one of successive approximations. Initially the answer is assumed to be at the midpoint of the possible region (00 to 128). The guess is then squared and compared to the number. Using the result of the comparison, the region of possible answers is divided by two by setting the guess to be a new maximum or minimum value. The next guess is chosen to be at the half way point of the new region. This process continues until the region has been reduced to a single integer. This number is then returned in the accumulator.

Before any distances can be calculated, the coordinates must be expressed as a single integer.

Initially each coordinate is stored as three ASCII characters. Subroutine PACK takes the three characters pointed to by the HL register and converts the number to a single byte. In order to accommodate the fractional part of the coordinate, the answer is multiplied by two before it is returned in the accumulator.

The distance between any two transmitters can be determined with the DIST subroutine. This subroutine utilizes the PACK subroutine to obtain the coordinates of each transmitter. The distance is then calculated using the Pythagorus Theorem. The result is returned in register A.

The DISTA subroutine calculates the distance between the first transmitter and the transmitter whose number is entered in the accumulator. The DIST subroutine is employed to do the calculations.

It is sometimes desireable to increase the cassette store rate when a transmitter gets within a certain distance (DISTC) of the first transmitter. CSINC controls the cassette store rate. The DISTA subroutine is used to calculate the distance between a transmitter and the first transmitter. If one of the distances is less than DISTC, the cassette store rate is increased to the scan rate. If the activity state of the transmitter that is within range is zero, the rate is not increased.

Setting the Flags

The activity flags are set by the ACTAD subroutine and then overidden by the routine which loads the data into the cassette buffer (LOADC). The LOADC subroutine is discussed later.

The test antenna error flags are set by the STTAN subroutine. This subroutine compares the readings on the test antenna with a comparator value (TANCP). If the reading for a particular channel is greater than the comparator value, the flag is set, otherwise it is reset.

STABL sets the above/below flag. This flag is intended to indicate whether the transmitter is above or below the snow. Since this feature is not yet implemented, a zero is stored for these flags.

Loading the Cassette

A single line of data from the temporary buffer can be transferred to the cassette buffer by the LDCAS subroutine. The number corresponding to the transmitter data to be loaded is entered in the accumulator. This subroutine first loads a group separator, then the data, and finally a carriage return.

The subroutine LOADC decides whether or not data for a particular transmitter are sent to the cassette buffer. This subroutine is called every cassette store interval. The routine compares the position and activity of the transmitter with its position and activity for the last cassette store interval. If any data have changed, the data for that transmitter are loaded into the cassette buffer by calling LDCAS. In order to store the data for all possible transmitters, the LODEC subroutine calls LOADC once for each active channel.

Loading the Printer

In order to print the data, it must be loaded into the print buffer. The interrupt driven printer routine (PTINT) is then called. LDPRT transfers the data from the temporary buffer into the print buffer. The ACLOC flag is first checked to determine whether the locations or the flags should be printed. If the positions are to be printed, the real time, the channel numbers, and the X and Y coordinates are loaded into the print buffer. The coordinates are followed by the carriage control characters. If the flags are to be printed, they are transferred to the print buffer instead of the position coordinates.

Utility Subroutines

Three utility subroutines are also included in this section. INCD5 increments the DE register five times, while IHL20 increments the HL register twenty times. A general purpose pointer incrementing subroutine (INPNT) is also included. Pointers to be incremented in multiples of 1 are loaded into I1T1 to I1T5. Pointers to be incremented in multiples of 22 are entered in I22T1 to I22T3. Register A is then loaded with the number of times the routine should increment the variables. For example, if register A is loaded with 3, the variables in I1T1 to I1T5 are incremented by 3, while the variables in I22T1 to I22T3 are incremented by 66.

V. SUMMARY

Initial testing demonstrates that the data aquisition system successfully monitors the antennas and locates a transmitter placed under the grid. All features of the system operate as expected.

Although the system functions as planned, modifications could be made to increase its versatilty. Much of the software required to produce an above/below the snow flag is already incorporated into the system. In order to make this feature operational a program has to be added which would process the data to determine whether the transmitter is above or below the snow. The subroutines to set/reset the flag and load the flag onto the cassette tape or printer are included in the present software.

Keyboard entry of commands adds greatly to the versatility of the system. Any number of commands can be added to modify the operating procedure of the system. Commands are already included which allow the user to modify the activity algorithm or the interpolation algorithm.

Some of the final data processing could also be implemented in this system. This includes such features as calculating the home range of animals, velocities of movement, etc. In most cases, however, this processing is best done on a general purpose computer where the required programs are more easily developed. All of the data required for many of these features are already included on the cassette tape.

Besides the specific application considered here, the system could be modified for many other applications. The size of the grid could be increased to allow for tracking larger animals. With minor modifications to the program, the system could also monitor a different arrangement of antennas.

The system can also function as a voltmeter. If the signal from the antenna control box is sent directly to the computer, instead of through the radio, the voltage of any line connected to one of the relays can be read by the analog to digital converter. The system would then become a 42 input programmable digital voltmeter.

Although the system operates properly, several improvements could be made. Since the printer is a mechanical device, it is likely that this is the part of the system that will cause the most trouble. A better approach would have been to supply the computer with a standard RS-232 computer terminal interface. Any commercial terminal could then be connected to the system and used for the interface to the user. This approach would, however, be more expensive unless a terminal was already available. If a video terminal was connected to the system in this manner, many other features could be easily added. A grid on the video screen could, for example, display the current position of each transmitter by placing the transmitter number at the corresponding grid position on the screen.

REFERENCES

ALLEN, D.M. Saturation recording is not that hard. Byte. Vol. 2 No. 1 p. 34, 1977.

BIDER, J.R. Animal activity in uncontrolled terrestrial communities as determined by a sand-transect technique. Ecol.Mongr. Vol. 38 p. 269,1968.

CHUTE, F.S., FULLER, W.A., HARDING, P.J.R., HERMAN, T.B. Radio tracking of small mammals using a grid of overhead wire antennas. Can. J. Zool. Vol. 52 p. 1481, 1974.

GUZMAN, D. Marry your microprocessor to monolithic A/D's. Electronic Design. Vol. 25 No. 2 p. 82, 1977.

HELMERS, C. A software controlled 1200 BPS audio tape interface. Byte. Vol. 2 No. 4 p. 40, 1977.

JUSTICE, K.E. A new method for measuring home ranges of small mammals. J.Mammal. Vol. 42 p. 462, 1961.

KINZER, D. A simple cassette interface. IEEE Circuits and Systems. Vol. 11 No. 1 p. 18, 1977.

KOLZ, A.L., CORNER, C.W., TIETJEN, H.P. A radio frequency beacon transmitter for small mammals. J. Wildl. Manage. Vol. 36 p. 177, 1972.

LEWIS, T.G., DOERR, J.W. Minicomputers: Structure and Programming. Hayden Book Co. 1976.

MANLY, W.A. Magnetic recording for computers. The Best of Byte. Vol. 1 p. 44, 1977.

MAUCH, H.A. Digital data on cassette recorders. The Best of Byte. Vol. 1 p. 184, 1977.

MILLER, L.S. Tracing vole movement by radioactive excretory products. Ecology. Vol. 38 p. 132, 1957.

MOTOROLA Application Note AN-711. The recovery of recorded digital information in drum, disk and tape systems.

PRUITT, W.O., Jr. A method of live-trapping small taiga mammals in winter. J.Mammal. Vol. 40 p. 139, 1959.

RAMPIL, I.,BREIMER, J. The Digital Cassette Subsystem. Byte Vol. 2 No. 2 p. 24 and Vol. 2 No. 3 p. 38, 1977.

RAWSON, K.S., HARTLINE, P.H. Telemetry of homing behavior by the deermouse, Peromyscus. Science Vol. 146 p. 1596, 1964.

SCHMID, H., MROZOWSKI, G. Mating microprocessors with converters. Electronic Design. Vol. 23 No. 18 p. 76, 1975.

SUDING, Dr. Robert. Why wait? Build a fast cassette interface. The Best of Byte. Vol. 1 p. 190, 1977.

VERMEULEN, H. Beverage antennas for biotelemetry. M.Sc. Thesis, Department of Electrical Engineering, University of Alberta.

APPENDIX A: User's Guide

The data acquisition system scans the antennas every scan interval and checks the positions of any transmitters located within the grid. A maximum number of five transmitters may be used at any time. The data collected can then be displayed on the printer and stored on the cassette tape. Data are stored on the cassette tape to facilitate processing at a later time.

When the mode switch is set to CONTROL, the keyboard becomes active, and commands can be entered into the computer. Before using the system, it is necessary to initialize the various parameters in the system, such as the channels that are being used, the correct time, the scanning interval, and various other options which may be chosen. When the computer is reset, many of the system parameters are set to a default value. If these values are not suitable, they can be changed by issuing appropriate keyboard commands.

Once the system has been initialized, the mode switch is set to SCAN. The keyboard no longer has control of the system, and the data acquisition procedure is started. As data are collected, they are stored in a buffer memory until there are enough data to be output to the cassette tape. When the buffer is full, the cassette drive is automatically turned on, and the data are dumped onto the tape.

Besides the position of each transmitter, the activity state is also recorded. An activity state of 1 indicates a transmitter which is in motion. A motionless transmitter has an activity state of 0.

A test antenna is also checked to determine the level of radio interference in the area. If the level on the test antenna is greater than a selected level, a flag is set to indicate that the data may be in error.

The printer output is in the following format:

TTTTTT CC XXX YYY

or

TTTTTT CC ABT

where TTTTTT is the time (hr,min,sec)

CC is the channel number

XXX is the position on the X axis (antenna XX.X)

YYY is the position on the Y axis (antenna YY.Y)

A is the activity flag with a 0 signifying a motionless transmitter, and a 1 indicating a transmitter in motion.

B is a flag indicating whether the transmitter is above or below the snow. This feature is not yet implemented.

T is the test antenna flag. A 1 indicates that the radio interference is above the predetermined level.

The output to the cassette is in the following format:

TTTTTT CC XXX YYY ABT

There are three rates at which the system performs various tasks. The scan rate is the rate at which the system scans the antennas and calculates the positions of all transmitters. The operation of the printer is controlled by the print interval. Every print interval the data for all active transmitters is printed out. The cassette store rate is the rate at which data can be sent to the cassette buffer. If the cassette rate is set lower than the scan rate, not all data collected are necessarily stored on the cassette tape.

The cassette system checks the data every cassette store interval. If the position or activity state for a particular transmitter has changed in the last cassette store interval, the new data are sent to the cassette buffer.

If the operator chooses, one transmitter may be selected as the master transmitter. If any of the other transmitters come within a specified distance of this master transmitter, the cassette store rate is increased to the scanning rate. When the cassette store rate is the same as the scan rate, all data collected are stored on the tape. The cassette store rate returns to its previous value when the transmitters move apart, or if no activity is detected on the transmitter which is within range.

Before a cassette tape can be removed, an end of file character must be stored on the tape so that the receiving system knows that there is no more data to be read. The procedure for closing the cassette tape file varies depending on whether or not you wish the system to continue scanning.

Detailed procedures for controlling the various parts of the system are now given. A complete description of each command is given in the section "Keyboard Commands".

1. Initialization
   - a. Put the mode switch on CONTROL and turn on the power. The message 'SYSTEM RESET' should then be printed.
   - b. If the reset message is not printed, depress the reset switch. The message will then be printed.
   - c. The system is now reset and all internal parameters are set to their default values. The system is now in control mode ready to accept keyboard commands. The control mode is indicated by a prompt character (#) at the start of the line.
   - d. If any of the parameters are to be changed from their default values, enter the appropriate commands via the keyboard. (see the section "Control Parameters")
   - e. The system is now initialized and ready to start operation.
2. Control Parameters

   Before entering the scan mode, several parameters must be initialized.
   - a. Use the TST command to set the real time clock. The real time clock allows the correct time to be output to the printer and cassette tape.
   - b. Use the SAC command to enter the channels which are used. If a master transmitter is to be used, it is selected by entering it first. The first three

channel numbers entered are the ones for which population data are collected. (see the section "Population Matrix Facility")

c. If the cassette rate should increase when a transmitter gets within a certain range of the master transmitter, this function must be initialized via the DIS command.

d. If the system is to scan at a different rate than the default value, the SCI command is used to set the new scan interval.

e. If the printer is to display the transmitter data, it must be enabled and the print interval set via the PRI command. The POF command disables the printer. If flags are to be printed instead of the positions, the FLG command is issued. The POS command returns the system to printing the positions.

f. If the multiplication constant is to be changed, see the section 'Interpolation Algorithm'. The multiplication constant is used to interpolate between antenna readings.

g. If the activity algorithm is to be altered, the PCT command is issued. (see the section 'Activity Algorithm')

h. The setpoint for the test antenna can be changed from its default value with the TAN command.

i. The cassette interface must also be initialized before the system is operational. (see the section "Cassette Operation")

j. The gain control must be set to obtain optimum results. (see the section "Setting the Gain Control")

k. Before entering the scan mode, use the DSP command to make sure that all parameters have been set as requested. DSP ALL displays all parameters which have been changed from their default values.

l. Set the mode switch to SCAN to start the scanning procedure.

An example of setting various parameters is now given. User input is underlined.

```
#TST13:45<space>
#SAC 05,07,14.
#PRI 05
#DSP ALL
TIME IS 13:46:17
PRINTER ENABLED
PRINT INT.= 05
ACTIVE CHANNELS
05,07,14
#
```

The user first sets the clock to the correct time. The channel numbers for the transmitters to be used are entered. The printer is then enabled, and the print interval is set to 5. Since these parameters have been changed from their default values, they are displayed by the DSP ALL command.

3. Setting the Gain Control
    a. Place a transmitter at the corner of the grid closest to the receiver.
    b. Orient the transmitter for maximum coupling onto one antenna (X00 or Y00).
    c. Use the CHN command to set the receiver to the channel corresponding to the transmitter used.
    d. Use the ANT command to switch in the antenna chosen in step b.
    e. Switch on the meter and adjust the gain control until the meter reads approximately 8 to 10 volts.
    f. Be sure to turn off the meter before entering the SCAN mode.
4. Cassette Operation

Whenever the mode switch is set to SCAN, the data collected are stored in a buffer memory until there is enough data to ouput to the tape. The cassette drive is then started and the contents of the buffer memory are dumped onto the tape. If the experiment to be run is expected to use more than one side of a cassette, it is suggested that enough cassettes be initialized before entering the scan mode. Having these cassettes available makes it easier to change tapes without interrupting the scanning procedure.

a. Cassette Initialization

1) Before starting to record data on a tape, the tape must be rewound. Use the CON command to apply power to the cassette drive.
2) The tape in the cassette must now be moved until the entire leader is moved onto the take-up reel. Use the PLAY switch to move the tape until there is actual recording tape on the reel.
3) Now that the tape is ready, use the COF command to turn off the cassette drive.
4) Depress the PLAY switch. The cassette drive is now ready to receive data.
5) If a non standard tape length is used, use the CTP command to change the tape length from its default value.
6) When the mode switch is set to SCAN, the cassette drive is operational.

b. Removing Cassettes

If there is no more data to be placed on the tape, the following steps should be followed to remove the cassette:

1) Move the mode switch to CONTROL.
2) If the CASSETTE ON indicator is on, wait for it to turn off.
3) Use the STP command to close the cassette file.
4) Wait for the CASSETTE DONE indicator to turn on.

5) The cassette can now be removed.

c. Changing Cassettes While Scanning

If the cassette tape becomes full of data, the alarm is sounded. The following procedure should be followed to ensure that no data are lost.

1) Depress the CASSETTE RESET switch.
2) Wait for the CASSETTE DONE indicator to light up.
3) When the indicator is lit, the cassette can be removed and a new cassette inserted. The new tape must be inserted, and the drive returned to the play mode, before the CASSETTE DONE indicator turns off.
4) The new tape must have been previously rewound and the leader moved onto the take-up reel. (see section "Cassette Initialization")

d. Entering Text Onto Tape

Text can be entered onto the tape via the CPR command. After the CPR command is entered, the printer responds with a carriage return and a line feed. Text can now be entered from the keyboard.

To obtain a carriage return and a line feed, type <CR> (CNTRL M). In order to make a correction type <BS> (CNTRL H), this command will result in a backspace. When the text has been entered, type <ETX> (CNTRL C) to return to the monitor.

If the <DEL> key is depressed before <ETX> the entire text is erased.

Since this command enters the text into the buffer memory, the text is not entered onto the tape until the next tape dump. The text is therefore lost if the system is reset (CASSETTE RESET or master RESET) before the next tape dump.

The command PRT acts similiarily to CPR, but in the case of PRT the text is entered only onto the printer.

5. Population Matrix Facility

The population matrix routines automatically count the number of times a transmitter has been found at a particular location. Every scan interval, the counter corresponding to the location of each transmitter is incremented by one. The only channels for which these data are collected are the first three channels entered via the SAC command.

a. The CLR command clears the entire population matrix.

b. The PPM command prints the population matrix for a particular transmitter.

6. Interpolation Algorithm

The signal level from each antenna is stored in memory in preparation for processing. The first step is to scan through these readings to find the maximum value. Suppose that antenna AA has the largest signal. This signal level is then compared to the neighbouring antennas in the following manner. (AA)*25 is compared to

(AA-1)*MLC and (AA+1)*MLC. If either one is greater than (AA)*25, the transmitter is said to be between the two antennas. MLC is a constant which can be set by the operator. For example if (AA+1)*MLC > (AA)*25 then the transmitter is said to be at AA.5. The constant MLC can be changed from its default value with the MLC command.

7. Activity Algorithm

Whenever a transmitter does not change locations between scan intervals, the system tries to determine whether or not the transmitter is moving. The system checks for motion by comparing successive signal levels from the closest antenna. The variable PCT contains a percentage value, say PP%. If the signal on the antenna varies by more than PP% between scan intervals, the transmitter is assumed to be moving, and the activity flag is set to one. If the signal level varies by less than PP% for a period of five consecutive scan intervals, the transmitter is said to be motionless, and the activity flag is set to zero.

8. In Case of Failure

If the system fails to operate, check the power supplies. The indicator lights are located near the main power switch. There is a light corresponding to each of the four supplies. If all four lights are out it is likely that the main fuse has blown. The fuses can be located behind the top panel on the front of the computer. (the fuses to be used are 6 Amp fast blo)

If either the +15 or +5 supplies are off, try changing the appropriate fuse.

If either the -15 or -5 supplies are off, or if changing the fuses as suggested above does not help, the system requires servicing.

KEYBOARD COMMANDS

Several of the system functions operate differently on different transmitter numbers. The transmitter number is set when the active channels are set via the SAC command. The first channel entered corresponds to transmitter one, the second to transmitter two, etc. The master transmitter is defined as transmitter one.

System Commands

DIS XX cassette rate increase distance

where XX=the range (in antennas) about the master transmitter

Whenever a transmitter comes within XX antennas of the master transmitter, the cassette store interval is increased to the scan interval. The cassette store rate returns to its previous value when the transmitter moves out of range, of if no activity is detected on the transmitter which is within range.

The default is zero. If the distance is set to zero, the program never increases the cassette store rate.

DSP AAA display programmable parameters

where AAA= the three character code for the parameter to be displayed

The parameters which can be displayed are CEN, CHN, CSI, CST, CTP, DIS, FLG, MIC, PCT, POS, PRI, SAC, SCI and TAN. The three character code

ALL results in the printing of all parameters which have been changed from their default values.

For example, DSP MLC causes the printer to print the current value of the multiplier constant.

MLC XX set multiplier constant for interpolation

where XX= multiplier constant

The variable used in the interpolation algorithm is set to the new value. For a discussion of the algorithm refer to the section "Interpolation Algorithm".

The default value is 50.

PCT XX set percentage for activity calculations

where XX= percentage

The percentage change in the signal levels, which determines whether or not the transmitter is moving, is set to the value entered. For a complete description of the algorithm used, refer to the section "Activity Algorithm".

The default is 20(%).

SCI XX set scan interval

where XX= time in seconds between scans

This routine allows the user to set the time interval between successive scans of the antenna system.

The default setting is 1 (second).

TAN XX test antenna setpoint

where XX= setpoint

Whenever the signal level on the test antenna is greater than the setpoint, the test antenna flag is set to 1.

The default is 20.

Cassette Interface Commands

CON turn on the cassette drive

Power to the cassette deck is turned on.

This command is only used for initializing the cassette tapes. In SCAN mode the system automatically turns on the cassette drive when data are to be stored on the tape.

COF turn off the cassette drive

Power to the cassette deck is turned off.

When in scan mode the power should be turned off since the scan routine turns on the power itself when it is sending data to the tape.

CSI XX set cassette store interval

where XX= the number of SCAN intervals between times when data are stored on tape

For example, if the cassette store interval is set to 5, and the scan interval is 2, the data are stored on tape every 10 seconds.

The default is 1. (every scan interval)

CTP XX set tape length

where XX= time in minutes of one side of the cassette tape

The system must know what size of tape is being used so that the alarm can be sounded when the tape becomes full.

The default is 30. (C60 cassette)

CPR put text onto tape

This routine allows text to be entered directly onto the cassette tape.

The computer first generates a carriage return and a line feed. Text can then be entered directly from the keyboard.

<CR> (CTRL M) generates a carriage return and line feed.

<BS> (CTRL H) generates a backspace.

<ETX> (CTRL C) returns control to the monitor.

<DEL> erases all text and returns control to the monitor.

STP close cassette tape file

The cassette drive turns on and the end of file character is loaded onto the tape. When finished, the cassette drive turns off and the CASSETTE DONE indicator turns on. The CASSETTE DONE indicator stays on until the next time the cassette drive is used.

Printer Control Commands

FLG print flags instead of positions

The printer prints the flags instead of the transmitter positions.

The default is to print the locations.

POF disable the printer

No data are output to printer.

Upon reset the printer is disabled.

POS print positions instead of flags

The printer prints the transmitter locations instead of the flags.

This is the default condition.

PRI XX set printer interval

where XX= print interval (no. of cassette save intervals)

This routine enables the printer, and sets the print interval. The print interval gives the number of cassette save intervals between successive printings.

e.g. If the scan interval is 2 and the cassette save interval is 3, and the command PRI 04 is entered, the transmitter data are printed every 24(2*3*4) seconds.

Upon reset the printer is disabled.

PRT put text onto printer

This routine allows text to be entered directly onto the printer.

The computer first generates a carriage return and a line feed. Text can then be entered directly from the keyboard.

<CR> (CTRL M) generates a carriage return and line feed.

<BS> (CTRL H) generates a backspace.

<ETX> (CTRL C) returns control to the monitor.

Population Matrix Commands

CLR clear population matrix

All values in the population matrix are set to zero.

PPM XX print population matrix

where XX= the transmitter number (1 to 3)

The entire population matrix for transmitter XX is printed.

Radio Control Commands

SAC AA,BB,CC,DD,EE,FF set channels to be used

where AA,BB,...= channel numbers to be used

This routine sets the channels to be used by the scan routine. The channels are scanned in the order entered.

If there are less than five channels to be used, enter a period (.) in place of the comma (,) after the last channel entered.

For example, SAC 01,05,27. results in the system checking for signals on channels 1, 5, and 27.

Gain Control Commands

ANT AZZ select antenna

where A= X or Y

ZZ= antenna number (00 to 22)

The computer first selects the appropriate relay, and then reads the signal level. The response is:

```
ANT AZZ=X.X VOLTS
```

The value printed should be approximately half that of the meter reading.

CHN XX set radio channel

where XX= channel number

The radio is set to channel XX.

Real Time Clock Commands

TST XX:YY set real time clock

where XX= time in hours

YY= time in minutes

The clock starts when the space bar is hit.

TIM display time

The time is printed.

Maintenance Commands

ALM turn on the cassette alarm

The alarm can be turned off with the cassette reset switch.

CEN XXXX set end of cassette buffer memory

where XXXX= ending memory address (hex)

This routine is used along with CST to set the region of memory used for the cassette buffer.

The default is 3FFF.

CST XXXX set start of cassette buffer memory

where XXXX= starting memory address (hex)

This routine is used along with CEN to set the region of memory used for the cassette buffer.

The default is 3000.

CIN initialize cassette tape system

The cassette buffer is emptied and the timer for the alarm is set for the selected tape length.

DPM XXXX YYYY display memory

where XXXX= starting memory address (hex)

YYYY= ending memory address (hex)

The addresses and data are both printed in hexadecimal.

INP XX read input port

where XX= port number (hex)

The data from the port are printed in hexadecimal.

OUT XX ZZ output data to port

where XX= output port number (hex)

ZZ= data byte (hex)

The data (ZZ) are output to the specified port.

RUN XXXX. run program

where XXXX= address where excecution is to begin (hex)

Program excecution starts at address XXXX.

STM XXXX set memory locations

where XXXX= starting memory address (hex)

Computer responds with:

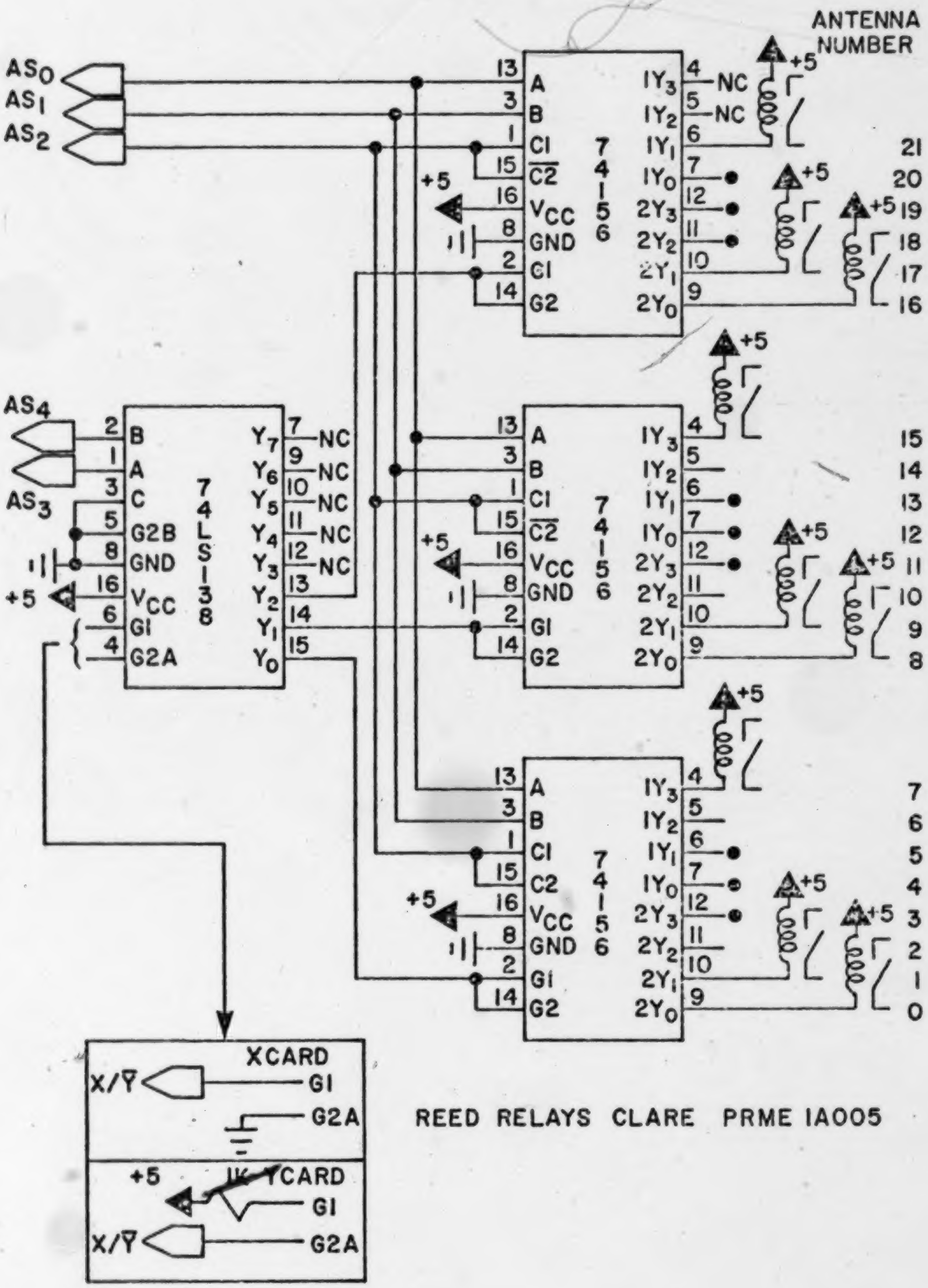
XXXX

The data byte (hex) is then inserted and the computer responds with the next sequential memory address:

XXXY

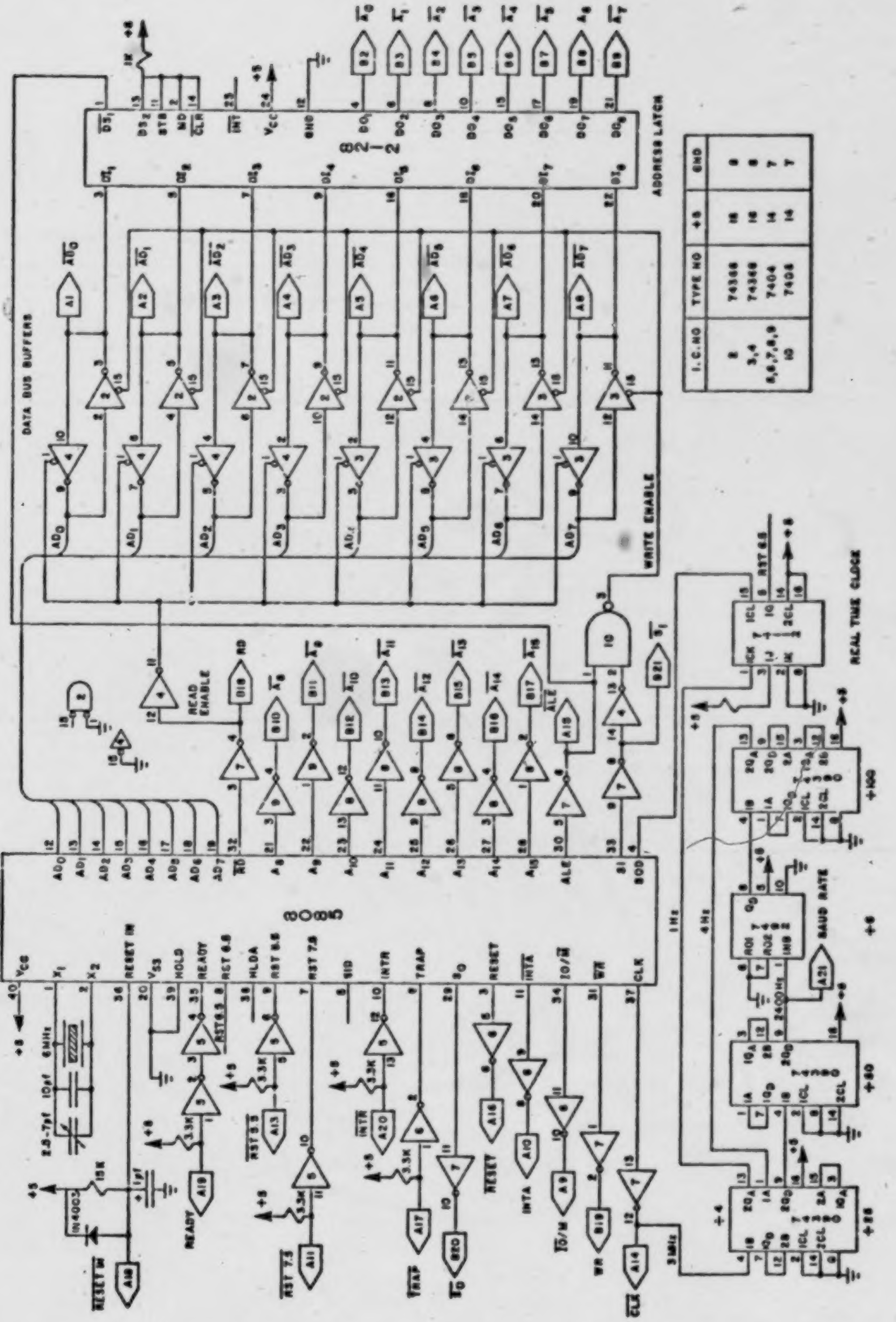
Another data byte can then be entered, or a period (.) to exit STM and return to the monitor.

APPENDIX B: Circuit Diagrams


ANTENNA NUMBER
AS0
AS1
AS2
AS3
AS4
74156
74LS138
13 A
3 B
1 C1
15 C2
16 VCC
8 GND
2 G1
14 G2
1Y3 4
1Y2 5
1Y1 6
1Y0 7
2Y3 12
2Y2 11
2Y1 10
2Y0 9
NC
+5
2 B
1 A
3 C
5 G2B
8 GND
16 VCC
6 G1
4 G2A
Y7 7
Y6 9
Y5 10
Y4 11
Y3 12
Y2 13
Y1 14
Y0 15
21
20
19
18
17
16
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
0
XCARD
X/Y̅
G1
G2A
~~1K~~ YCARD
+5

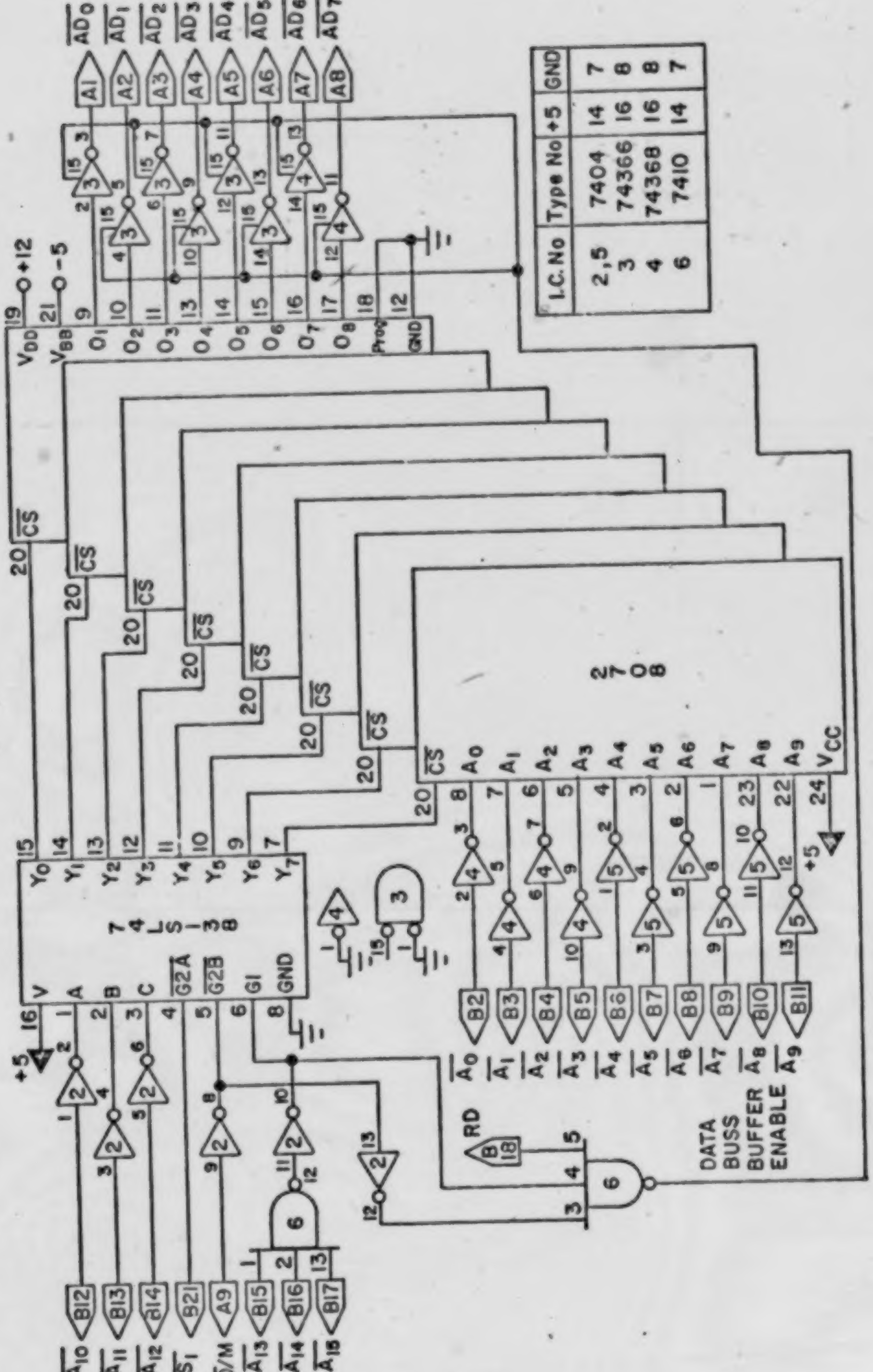
REED RELAYS CLARE PRME 1A005

Fig. 12 Relay Control Card


8085
8212
ADDRESS LATCH
DATA BUS BUFFERS
READ ENABLE
WRITE ENABLE
REAL TIME CLOCK
BAUD RATE
6MHz
3MHz
1Hz
4Hz
2400Hz
÷4
÷25
÷40
÷100
RESET IN
READY
HOLD
HLDA
TRAP
INTR
INTA
RESET
IO/M
WR
RD
CLK
ALE
SOD
SID
RST 5.5
RST 6.5
RST 7.5
AD0
AD1
AD2
AD3
AD4
AD5
AD6
AD7
IN4003
15K
3.3K
1K
+5

I.C. NO	TYPE NO	+5	GND
2	74368	16	8
3,4	74368	16	8
5,6,7,8,9	7404	14	7
10	7404	14	7

Fig. 13 Central Processor Card


$\overline{AD_0}$
$\overline{AD_1}$
$\overline{AD_2}$
$\overline{AD_3}$
$\overline{AD_4}$
$\overline{AD_5}$
$\overline{AD_6}$
$\overline{AD_7}$
A1
A2
A3
A4
A5
A6
A7
A8
+12
-5
V_{DD}
V_{BB}
O_1
O_2
O_3
O_4
O_5
O_6
O_7
O_8
Prog
GND
$\overline{CS}$
2708
A_0
A_1
A_2
A_3
A_4
A_5
A_6
A_7
A_8
A_9
V_{CC}
74LS138
Y_0
Y_1
Y_2
Y_3
Y_4
Y_5
Y_6
Y_7
V
A
B
C
$\overline{G2A}$
$\overline{G2B}$
G1
GND
+5
B2
B3
B4
B5
B6
B7
B8
B9
B10
B11
$\overline{A_0}$
$\overline{A_1}$
$\overline{A_2}$
$\overline{A_3}$
$\overline{A_4}$
$\overline{A_5}$
$\overline{A_6}$
$\overline{A_7}$
$\overline{A_8}$
$\overline{A_9}$
B12
B13
B14
B21
A9
B15
B16
B17
$\overline{A_{10}}$
$\overline{A_{11}}$
$\overline{A_{12}}$
$\overline{S_1}$
$\overline{IO}/M$
$\overline{A_{13}}$
$\overline{A_{14}}$
$\overline{A_{15}}$
RD
B18
DATA BUSS BUFFER ENABLE

I.C. No	Type No	+5	GND
2,5	7404	14	7
3	74366	16	8
4	74368	16	8
6	7410	14	7

Fig. 14 Program Memory Card

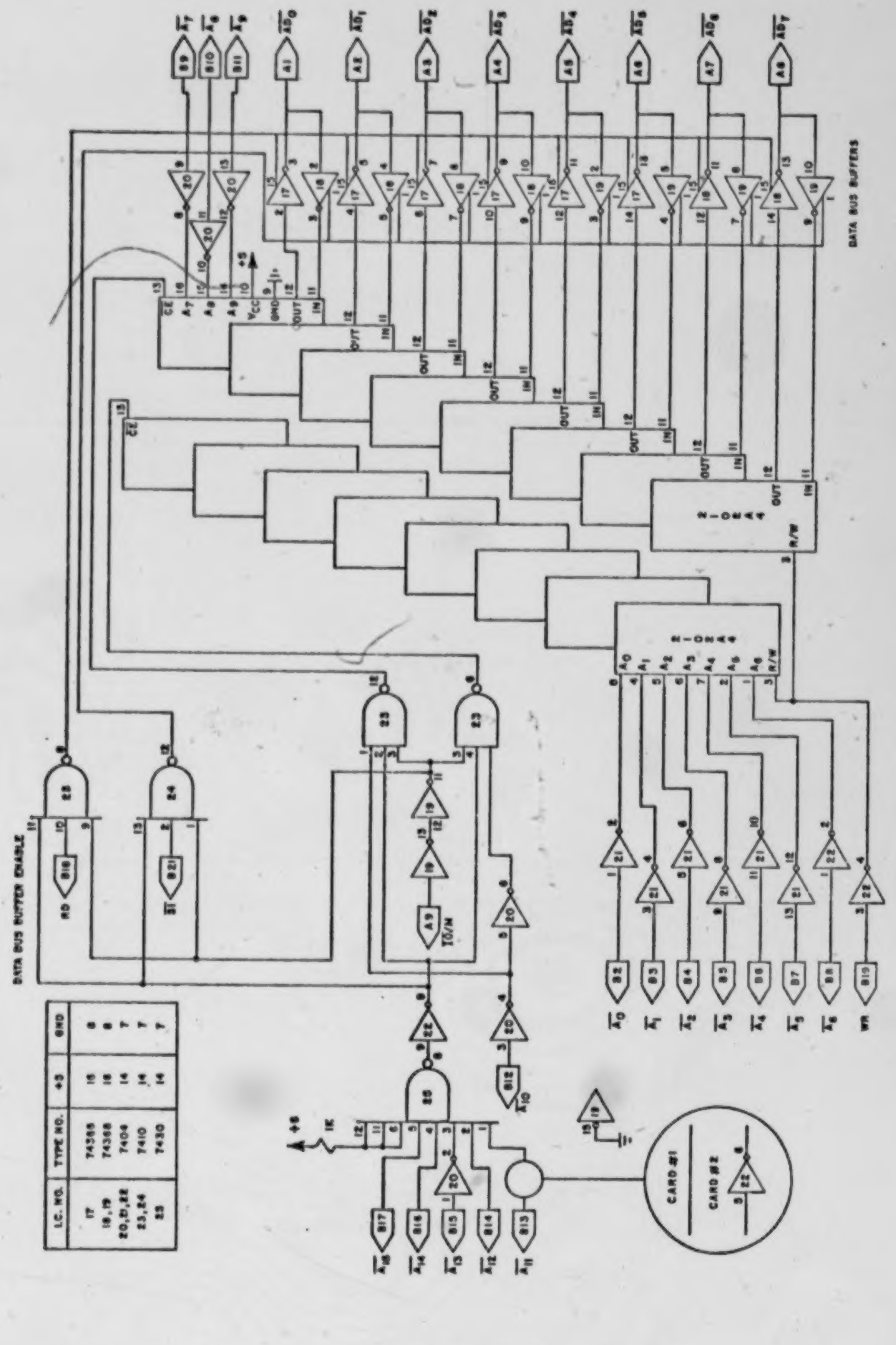

DATA BUS BUFFERS
DATA BUS BUFFER ENABLE
RD
B18
ST
B21
IO/M
A9
CARD #1
CARD #2
+5
1K
VCC
GND
OUT
IN
R/W
I.C. NO. | TYPE NO. | +5 | GND
17 | 74365 | 16 | 8
18,19 | 74368 | 16 | 8
20,21,22 | 7404 | 14 | 7
23,24 | 7410 | 14 | 7
25 | 7430 | 14 | 7

Fig. 15 Read/Write Memory Card

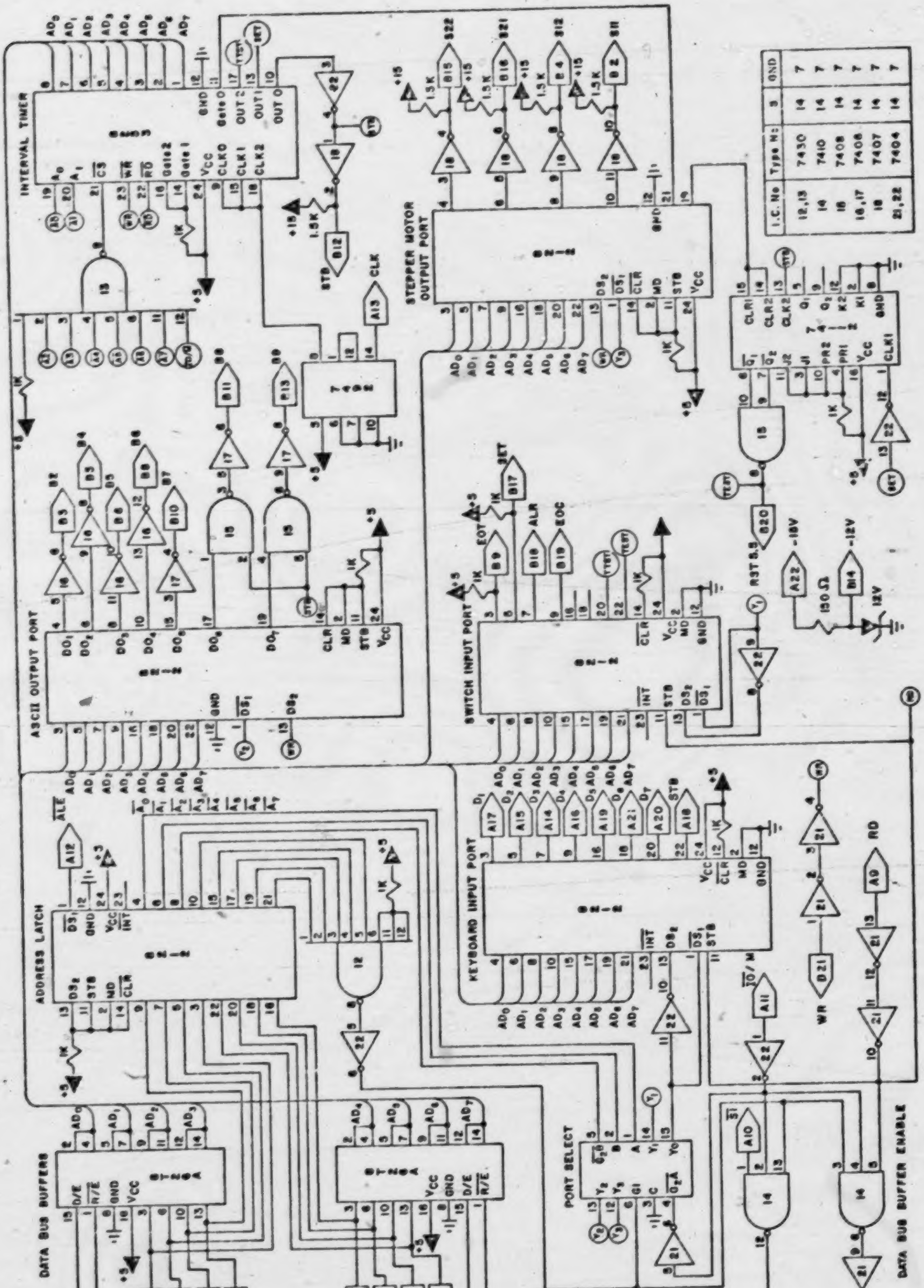

INTERVAL TIMER
STEPPER MOTOR OUTPUT PORT
ASCII OUTPUT PORT
SWITCH INPUT PORT
ADDRESS LATCH
KEYBOARD INPUT PORT
DATA BUS BUFFERS
PORT SELECT
DATA BUS BUFFER ENABLE
I.C. No | Type No | 5 | GND
12,13 | 7430 | 14 | 7
14 | 7410 | 14 | 7
15 | 7408 | 14 | 7
16,17 | 7406 | 14 | 7
18 | 7407 | 14 | 7
21,22 | 7404 | 14 | 7

Fig. 16 Keyboard/Printer Control Card

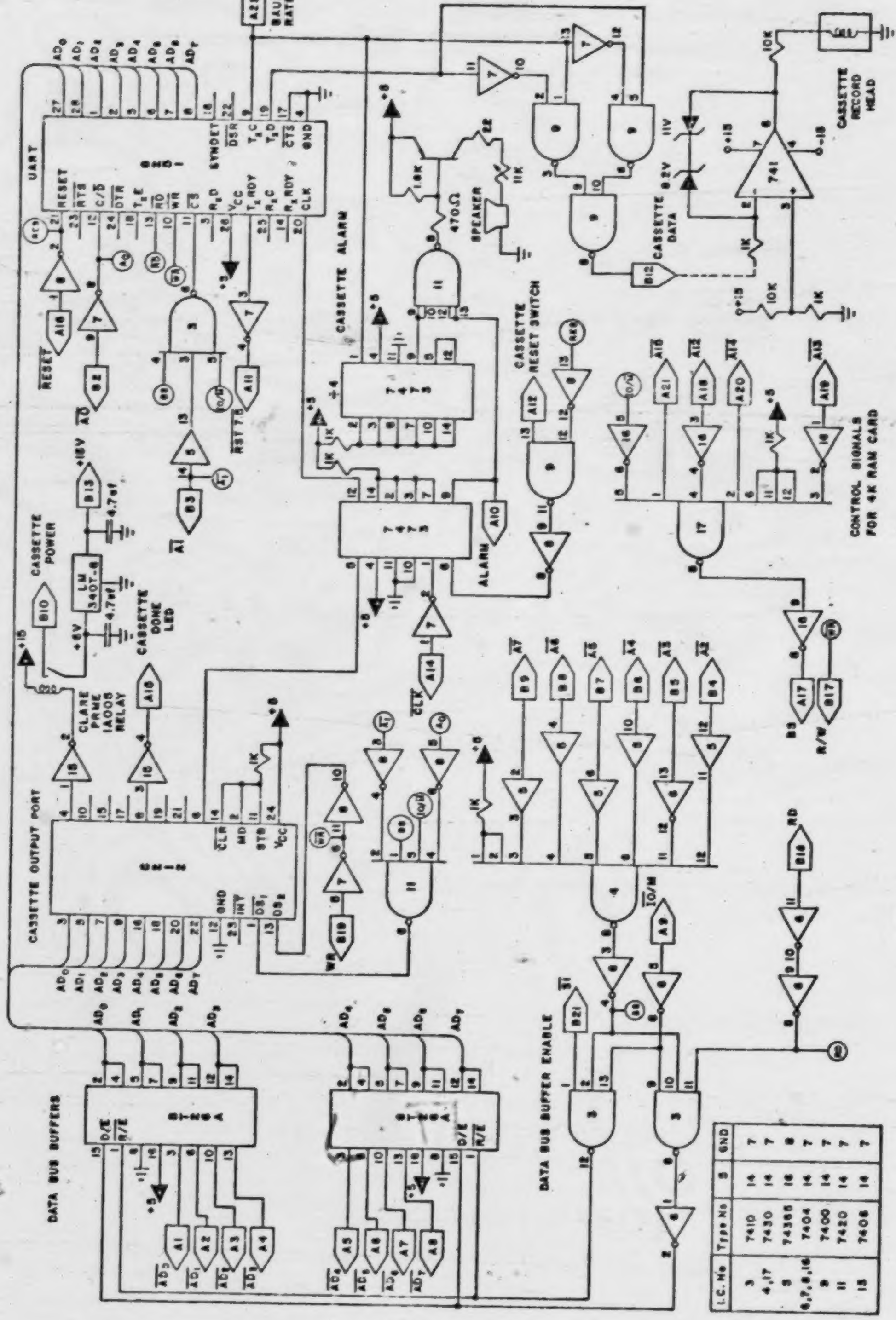
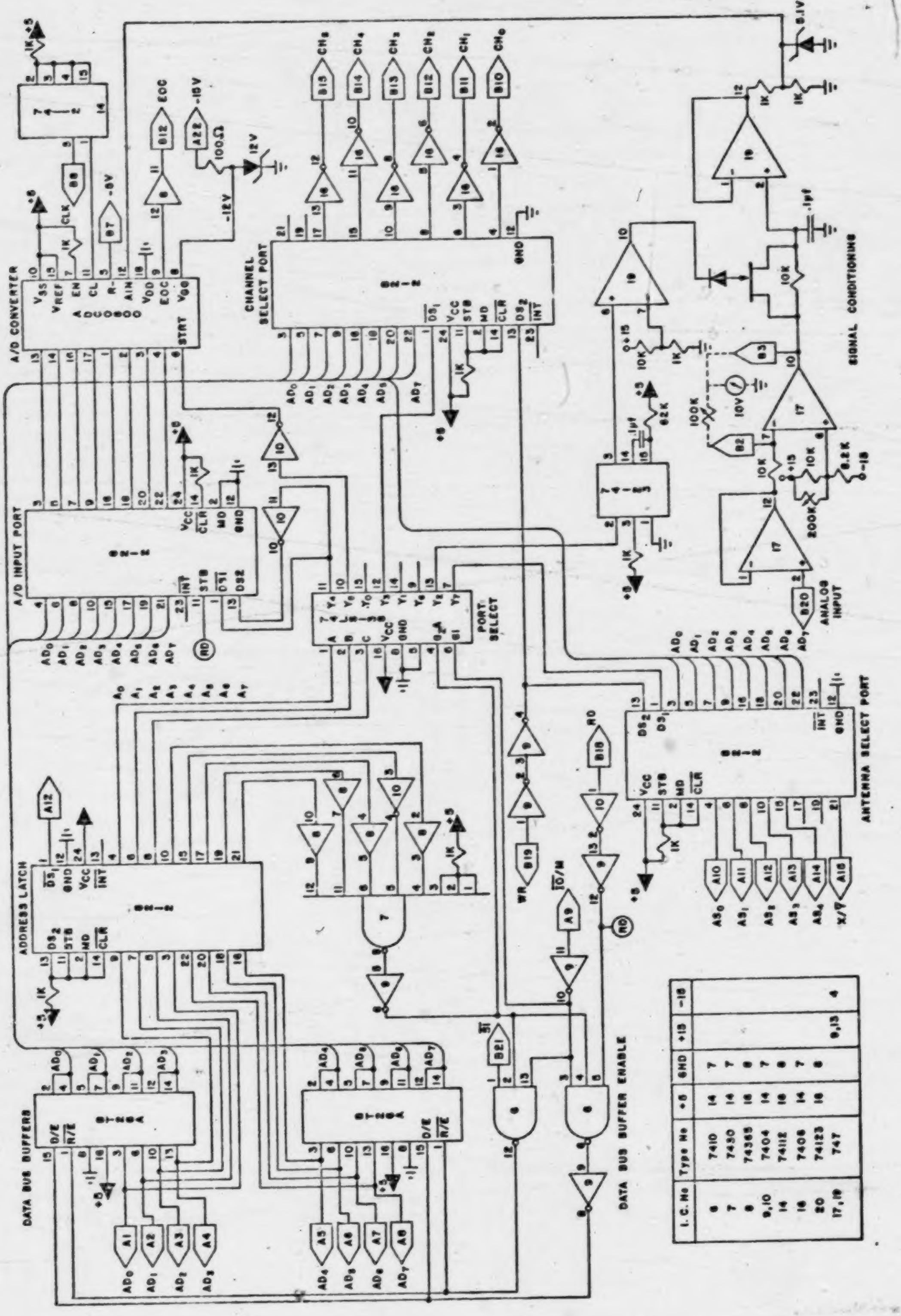
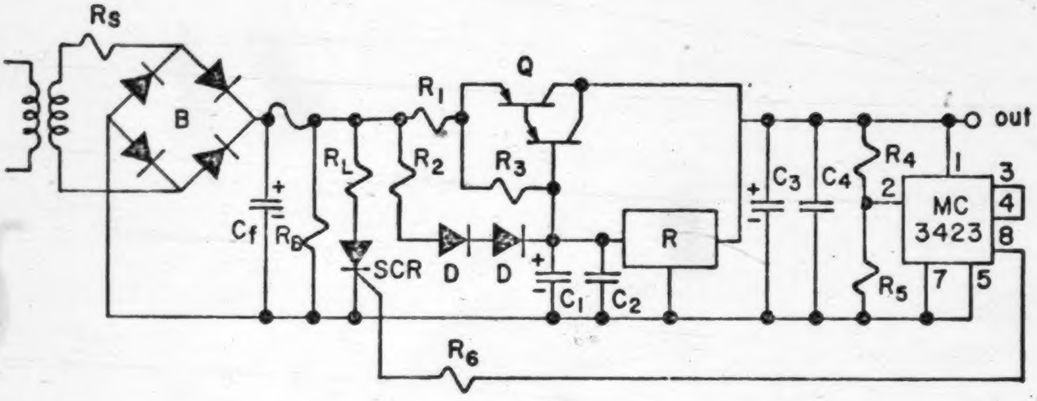

UART
8251
RESET
RTS
C/D
DTR
TxE
RD
WR
CS
RxD
VCC
TxRDY
RxC
RxRDY
CLK
SYNDET
DSR
TxC
TxD
CTS
GND
A22
BAUD RATE
CASSETTE ALARM
470Ω
SPEAKER
CASSETTE RESET SWITCH
CASSETTE DATA
8.2V
11V
741
CASSETTE RECORD HEAD
10K
1K
7473
÷4
ALARM
CASSETTE POWER
LM 340T-8
CASSETTE DONE LED
CLARE PRME 1A005 RELAY
CASSETTE OUTPUT PORT
8212
CLR
MD
STB
INT
DS1
DS2
CONTROL SIGNALS FOR 4K RAM CARD
R/W
DATA BUS BUFFERS
8T26A
D/E
R/E
DATA BUS BUFFER ENABLE
RD
WR
IO/M
CLK
RESET
RST 7.5
I.C. No | Type No | 5 | GND
3 | 7410 | 14 | 7
4,17 | 7430 | 14 | 7
5 | 74365 | 16 | 8
6,7,8,16 | 7404 | 14 | 7
9 | 7400 | 14 | 7
11 | 7420 | 14 | 7
15 | 7406 | 14 | 7

Fig. 17 Cassette Interface Card

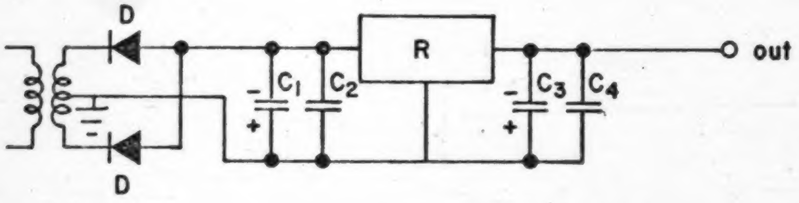

A/D CONVERTER
ADC0800
A/D INPUT PORT
CHANNEL SELECT PORT
PORT SELECT
ADDRESS LATCH
DATA BUS BUFFERS
DATA BUS BUFFER ENABLE
ANTENNA SELECT PORT
SIGNAL CONDITIONING
ANALOG INPUT
EOC
CLK
WR
RD
IO/M
5.1V
12V
-12V
-5V
-15V
100Ω
10V
100K
200K
82K
8.2K
10K
1K

I.C. No	Type No	+5	GND	+15	-15
6	7410	14	7		
7	7430	14	7		
8	74365	16	8		
9,10	7404	14	7		
14	74112	16	8		
16	7406	14	7		
20	74123	16	8		
17,19	747			9,13	4

Fig. 18 Antenna/Radio Control Card


Rs
B
Q
R1
RL
R2
R3
Cf
RB
SCR
D
D
C1
C2
R
C3
C4
R4
R5
R6
MC 3423
1
2
3
4
8
7
5
out

+5 Volt 5Amp				+15 Volt 5 Amp			
R	LM309			R	LM 340-15V		
B	IR250MB2-773	R_S	.25‖.25 100W	B	IR250MB2-773	R_S	.5 25W
SCR	ECC S2035	R_B	1K	SCR	ECC S2035	R_B	1K
D	1N4001	R_1	.2Ω 50W	D	1N4001	R_1	.1Ω 5W
Q	2N6287	R_2	5Ω 13W	Q	2N6050	R_2	2Ω 13W
C_f	2 x 120,000 μf	R_3	33Ω	C_f	40,000 μf	R_3	22Ω
C_1	10 μf	R_4	1.8 K	C_1	10 μf	R_4	5.6 K
C_2	0.1 μf	R_5	1K	C_2	0.1 μf	R_5	1K
C_3	100 μf	R_6	100Ω	C_3	100 μf	R_6	820Ω
C_4	0.1 μf	R_L	.25Ω 50W	C_4	0.1 μf	R_L	1.5Ω 25W


D
D
C1
C2
R
C3
C4
out

-5V 1Amp				-15V 1Amp			
D	1N4001	R	LM320-5V	D	1N4001	R	LM320-15V
C_1	13,000 μf	C_3	4.7 μf	C_1	4,700 μf	C_3	4.7 μf
C_2	0.1 μf	C_4	0.1 μf	C_2	0.1 μf	C_4	0.1 μf

Fig. 19 Power Supplies

APPENDIX C: Program Listing

```
                  ;SWITCH NUMBERS
0001              EOT     EQU       01H
0002              SETSW   EQU       02H
0003              ALARM   EQU       03H
0004              EOC     EQU       04H
0007              TTST    EQU       07H
0008              TEST    EQU       08H
                  ;I/O PORTS
0000              MODEA   EQU       00H
0001              TIMA2   EQU       01H
0002              TIMA1   EQU       02H
0003              TIMA0   EQU       03H
0004              STPER   EQU       04H
0005              ASCII   EQU       05H
0006              SWCH    EQU       06H
0007              KEY     EQU       07H
0008              CASET   EQU       08H
0009              C8251   EQU       09H
000A              CPORT   EQU       0AH
0010              ANTA    EQU       10H
0011              PORT    EQU       11H
0012              STCON   EQU       12H
0013              ADCIN   EQU       13H
0014              CHANL   EQU       14H
0015              FILTR   EQU       15H
                  ;RAM LOCATIONS
2000                      ORG       2000H
2000              CHR1:   DS        1
2001              CHR2:   DS        1
2002              CHR3:   DS        1
2003              CHR4:   DS        1
2004              CHR5:   DS        1
2005              CHR6:   DS        1
2006              CHR7:   DS        1
2007              CHR8:   DS        1
2008              CHR9:   DS        1
2009              CHR10:  DS        6
200F              CHR16:  DS        1
2010              CHR17:  DS        1
2011              CHR18:  DS        1
2012              POINT:  DS        2
2014              PRPOS:  DS        2
2016              SCAN:   DS        1
2017              SECO:   DS        5
201C              HR1:    DS        1
201D              CSTRT:  DS        2
201F              CEND:   DS        2
2021              NDAT:   DS        2
2023              OUTL:   DS        2
2025              BUFER:  DS        2
2027              CCNT:   DS        1
2028              CRDY:   DS        1
```

```
2029          CFIN:  DS       1
202A          OTEND: DS       2
202C          CASON: DS       1
202D          CASCT: DS       2
202F          TPLEN: DS       1
2030          ALRM:  DS       1
2031          CSCT:  DS       2
2033          EFILE: DS       1
2034          CNL:   DS       1
2035          CHNL:  DS       1
2036          CHN1:  DS       1
2037          CHN2:  DS       1
2038          CHN3:  DS       1
2039          CHN4:  DS       1
203A          CHN5:  DS       1
203B          TABLE: DS       210D
210D          TSTAN: DS       5
2112          TANCP: DS       1
2113          ANTNO: DS       1
2114          LOC:   DS       2
2116          PRB:   DS       2
2118          EOFF:  DS       1
2119          PRCNT: DS       1
211A          ACLOC: DS       1
211B          CNUM:  DS       1
211C          DISTC: DS       1
211D          ACPCT: DS       1
211E          ACPL1: DS       5
2123          ACPL2: DS       5
2128          ACCNT: DS       5
212D          ACPNT: DS       2
212F          I1T1:  DS       2
2131          I1T2:  DS       2
2133          I1T3:  DS       2
2135          I1T4:  DS       2
2137          I1T5:  DS       2
2139          I22T1: DS       2
213B          I22T2: DS       2
213D          I22T3: DS       2
213F          PRX1:  DS       30
215D          PRFL1: DS       5
2162          TM1:   DS       7
2169          CH1:   DS       3
216C          LCX1:  DS       4
2170          LCY1:  DS       4
2174          FLG1:  DS       4
2178          TM2:   DS       7
217F          CH2:   DS       3
2182          LCX2:  DS       4
2186          LCY2:  DS       4
218A          FLG2:  DS       4
218E          TM3:   DS       7
2195          CH3:   DS       3
2198          LCX3:  DS       4
```

```
219C        LCY3:  DS      4
21A0        FLG3:  DS      4
21A4        TM4:   DS      7
21AB        CH4:   DS      3
21AE        LCX4:  DS      4
21B2        LCY4:  DS      4
21B6        FLG4:  DS      4
21BA        TM5:   DS      7
21C1        CH5:   DS      3
21C4        LCX5:  DS      4
21C8        LCY5:  DS      4
21CC        FLG5:  DS      4
21D0        PRBF:  DS      95
222F        PROF:  DS      1
2230        PACT:  DS      1
2231        PINT:  DS      1
2232        PINT1: DS      1
2233        SCAN1: DS      1
2234        SCAN2: DS      1
2235        CASIP: DS      1
2236        CASI1: DS      1
2237        CASI2: DS      1
2238        APREV: DS      1
2239        ALAST: DS      1
223A        APRES: DS      1
223B        ANEXT: DS      1
223C        MAXNO: DS      1
223D        NEARY: DS      1
223E        NEARX: DS      1
223F        CMULT: DS      1
2240        AX:    DS      1
2241        AY:    DS      1
2242        BX:    DS      1
2243        BY:    DS      1
25A0               ORG     25A0H
25A0        POPM:  DS      2646
            ;CONSTANTS
0008        BAKSP  EQU     08H
000D        CRET   EQU     0DH
007F        DELET  EQU     7FH
0003        ETX    EQU     03H
001C        EOF    EQU     1CH
```

```
                ;********************************************
                ;*                                          *
                ;*        MACRO TO READ SWITCH INPUTS       *
                ;*                                          *
                ;*  THE PARAMETER LIST GIVES THE ACTUAL SWITCH*
                ;*NAME TO BE READ.                          *
                ;*  THE RESULT OF THE MACRO IS THE SETTING OF *
                ;*THE CARRY BIT IF THE SWITCH IS ON.        *
                ;*                                          *
                ;********************************************
  1             READ    MACRO   NAME
  1                     PUSH    D       ;SAVE DE REGISTERS
  1                     MVI  D, NAME    ;STORE SWITCH NUMBER IN D
  1                     IN      SWCH    ;READ THE SWITCH INPUTS
  1             NO:     RAR             ;ROTATE D0 INTO CARRY
  1                     DCR  D          ;DECREMENT COUNTER
  1                     JNZ     NO      ;NOT CORRECT SWITCH YET
  1                     POP     D       ;RESTORE DE REGISTERS
                        ENDM
                ;********************************************
                ;*                                          *
                ;*              INITIALIZATION              *
                ;*                                          *
                ;********************************************
0000                    ORG     0000H
0000 C34000             JMP     CONT
                ;  FORWARDING ADDRESSES FOR RST 6.5
0003 C38609             JMP     AD21
0006 C3E90C             JMP     PTINT
0009 C38D09             JMP     INDE
000C C3AE11             JMP     CRLF
000F C3B711             JMP     NLINE
0012 C34508             JMP     MULT8
0015 C37608             JMP     HXDEC
0018 C36408             JMP     DECHX
001B C39D08             JMP     CONHX
001E C3BC08             JMP     CONAS
0021 C3BC11             JMP     LEOF
0024 C39709             JMP     AD882
                ;  INTERRUPT VECTOR FOR RST 5.5 (PRINTER TIMER)
002C                    ORG     2CH
002C C3E90C             JMP     PTINT
                ;  INTERR. VECTOR FOR RST 6.5 (REAL TIME CLOCK)
0034                    ORG     34H
0034 C30018             JMP     1800H
                ;  INTERR. VECTOR FOR RST 7.5 (CASS. INTERFACE)
003C                    ORG     003CH
003C C36910             JMP     COUT
0040                    ORG     0040H
0040 319F25     CONT:   LXI  SP,259FH ;SET STACK POINTER
0043 AF                 XRA  A          ;CLEAR ACCUMULATOR
0044 D30A               OUT     CPORT ;TURN OFF CASSETTE
                ;  PRINTER INITIALIZATION
0046 D305               OUT     ASCII ;TURN OFF CHAR GENERATOR
```

```
0048 21B409          LXI  H, STEP
004B 221420          SHLD    PRPOS ;INITIALIZE PRPOS
004E CD000A          CALL    HOME  ;INITIALIZE PRINTHEAD
0051 CDB70A          CALL    CLPSK ;CLEAR THE PRINT STACK
               ;  DISABLE SCAN FLAG
0054 AF              XRA     A     ;CLEAR ACCUMULATOR
0055 321620          STA     SCAN  ;DISABLE SCAN FLAG
               ;  RESET REAL TIME CLOCK
0058 211720          LXI  H, SECO
005B 0606            MVI  B, 06
005D AF              XRA     A      ;CLEAR ACCUMULATOR
005E 77       CLEAR: MOV  M, A
005F 23              INX     H
0060 05              DCR     B
0061 C25E00          JNZ     CLEAR ;NOT ALL CLEARED
               ;  INITIALIZE UART (8251)
0064 3E79            MVI  A, 79H
0066 D309            OUT     C8251 ;SET MODE WORD
0068 3E10            MVI  A, 10H
006A D309            OUT     C8251 ;SET COMMAND WORD
               ;  SET SCAN INTERVAL
006C 3E01            MVI  A, 1
006E 323322          STA     SCAN1
0071 323422          STA     SCAN2 ;SET SCAN INTERVAL
               ;  SET CASSETTE STORE INTERVAL
0074 3E01            MVI  A, 1
0076 323522          STA     CASIP
0079 323622          STA     CASI1
007C 323722          STA     CASI2
               ;  SET MULTIPLIER CONSTANT
007F 3E32            MVI  A, 50
0081 323F22          STA     CMULT
               ;  SET COMPARATOR VALUE FOR TEST ANTENNA
0084 3E14            MVI  A, 20
0086 321221          STA     TANCP
               ;  SET PERCENTAGE FOR ACTIVITY CALCULATIONS
0089 3E14            MVI  A, 20
008B 321D21          STA     ACPCT
               ;  SET VALUE FOR SCAN RATE INCREASE
008E AF              XRA  A
008F 321C21          STA     DISTC
               ;  SET NUMBER OF CHANNELS
0092 321B21          STA     CNUM
               ;  SET ACTIVITY FLAGS
0095 3E31            MVI  A, 31H
0097 327421          STA     FLG1
009A 328A21          STA     FLG2
009D 32A021          STA     FLG3
00A0 32B621          STA     FLG4
00A3 32CC21          STA     FLG5
               ;  CLEAR POPULATION MATRIX
00A6 CD7A0F          CALL    SCLR
               ;  SET ACTIVITY COUNTERS
00A9 212821          LXI  H, ACCNT
```

```
00AC 0605                 MVI  B, 5
00AE 3E05                 MVI  A, 5
00B0 77          SAC5:    MOV  M, A
00B1 23                   INX  H
00B2 05                   DCR  B
00B3 C2B000               JNZ      SAC5
                 ;  SET DEFAULT VALUES
00B6 210030               LXI  H, 3000H
00B9 222520               SHLD     BUFER
00BC 221D20               SHLD     CSTRT
00BF 21FF3F               LXI  H, 3FFFH
00C2 221F20               SHLD     CEND
                 ;  NDAT=LINECOUNT * 2
00C5 212C00               LXI  H, 44
00C8 222120               SHLD     NDAT
00CB 219006               LXI  H, 1680D
00CE 222D20               SHLD     CASCT
00D1 223120               SHLD     CSCT
00D4 3E1E                 MVI  A, 30D
00D6 322F20               STA      TPLEN
00D9 3EFF                 MVI  A, 0FFH
00DB 322820               STA      CRDY
00DE AF                   XRA  A
00DF 321A21               STA      ACLOC
00E2 322C20               STA      CASON
00E5 323020               STA      ALRM
00E8 323320               STA      EFILE
00EB 322920               STA      CFIN
00EE 323520               STA      CHNL
00F1 322F22               STA      PROF
00F4 323022               STA      PACT
00F7 21F30C               LXI  H, START
00FA 221421               SHLD     LOC
                 ;  CLEAR CHANNEL NUMBERS
00FD 0605                 MVI  B, 05
00FF AF                   XRA  A
0100 213620               LXI  H, CHN1
0103 77          CL:      MOV  M, A
0104 23                   INX  H
0105 05                   DCR  B
0106 C20301               JNZ      CL
                 ;  PRINT RESET MESSAGE
0109 CD260A               CALL     RETN
010C CD9D0A               CALL     PRIMM
010F 53595354             DB       'SYSTEM RESET'
0113 454D2052
0117 45534554
011B 00                   DB       00H
011C CD000A               CALL     HOME
                 ;  ENABLE RST 6.5 FLIP FLOP
                 ;  ENABLE RST 7.5
                 ;  MASK RST 5.5
011F 3ED9                 MVI  A, 11011001B
0121 30                   DB       30H   ;SIM
```

```
0122 FB               EI               ;ENABLE INTERRUPTS
                  ;**********************************************
                  ;*                                            *
                  ;*                  MAIN PROGRAM              *
                  ;*                                            *
                  ;**********************************************
                  ;IF THE MODE SWITCH IS ON CONTROL THE PROCESSOR
                  ;JUMPS TO THE KEYBOARD ROUTINE TO ACCEPT
                  ;COMMANDS.  IF THE MODE SWITCH IS ON SCAN THE
                  ;SCAN FLAG IS SET.  THE SCAN FLAG IS USED TO
                  ;INDICATE TO THE REAL TIME CLOCK ROUTINE THAT
                  ;THE ANTENNAS SHOULD BE SCANNED AFTER THE CLOCK
                  ;IS INCREMENTED.  SUBROUTINES MOVI AND MOVR ARE
                  ;USED TO MOVE THE PRINTHEAD ONE POSITION TO THE
                  ;LEFT OR RIGHT RESPECTIVELY.
0123 1           +BEGIN: READ    SETSW ;READ SWITCH INPUTS
0123 1 D5        +       PUSH    D     ;SAVE DE REGISTERS
0124 1 1602      +       MVI  D, SETSW  ;STORE SWITCH NUMBER IN D
0126 1 DB06      +       IN      SWCH  ;READ THE SWITCH INPUTS
0128 1 1F        +NO:    RAR           ;ROTATE D0 INTO CARRY
0129 1 15        +       DCR  D        ;DECREMENT COUNTER
012A 1 C22801    +       JNZ     NO    ;NOT CORRECT SWITCH YET
012D 1 D1        +       POP     D     ;RESTORE DE REGISTERS
012E D23901              JNC     KEYB  ;JUMP TO KEYBOARD ROUTINE
                  ;  ENABLE SCAN FLAG
0131 3EFF                MVI  A, 0FFH
0133 321620              STA     SCAN
0136 C32301              JMP     BEGIN
                  ;**********************************************
                  ;*                                            *
                  ;*               KEYBOARD ROUTINE             *
                  ;*                                            *
                  ;*   THIS ROUTINE WILL ACCEPT AND STORE       *
                  ;* THREE CONSECUTIVE KEYBOARD CHARACTERS.     *
                  ;*   AFTER THE THIRD CHARACTER A JUMP TO THE  *
                  ;* CORRECT ROUTINE IS EXCECUTED.              *
                  ;*   REG E IS USED AS THE CHARACTER COUNTER   *
                  ;*   REG C IS USED AS A CHECK COUNTER         *
                  ;*                                            *
                  ;**********************************************
                  ;CHKEY WILL INPUT ONE CHARACTER FROM THE
                  ;KEYBOARD AND PRINT THE CHARACTER.
                  ;THE CHARACTERS ARE STORED IN THE PRINT STACK
                  ;ERROR IS CALLED WHENEVER THE KEYBOARD
                  ;ROUTINE DETECTS AN ENTRY THAT IS NOT A VALID
                  ;COMMAND. THIS ROUTINE PRINTS AN ERROR MESSAGE
                  ;ON THE PRINTER.
                  ;  IF CASSETTE IS JUST STARTING, WAIT FOR
                  ;  DRIVE TO REACH PROPER SPEED.
0139 3A2820       KEYB:  LDA     CRDY  ;CHECK FOR CASSETTE ON
013C A7                  ANA  A        ;SET FLAGS
013D CA3901              JZ      KEYB  ;WAIT FOR CASSETTE
                  ;  CASSETTE IS READY, CONTINUE WITH PROGRAM
0140 319F25              LXI  SP,259FH ;SET STACK POINTER
```

```
                    ;  COMPUTER IS IN CONTROL MODE, THEREFORE
                    ;  DISABLE THE SCAN FLAG.  THE REAL TIME CLOCK
                    ;  ROUTINE WILL THEN NOT SCAN THE ANTENNAS.
0143 AF                  XRA     A     ;CLEAR ACCUMULATOR
0144 321620              STA     SCAN  ;DISABLE SCANNING
                    ;  IF THE PRINTER IS PRINTING DATA
                    ;  WAIT FOR PRINTER TO FINISH
0147 3A3022              LDA     PACT  ;PRINTER IS ACTIVE
014A A7                  ANA  A        ;SET FLAGS
014B C23901              JNZ     KEYB  ;PRINTER STILL ACTIVE
                    ;  PRINTER IS NOT PRINTING, INITIALIZE
                    ;  THE PRINTER FOR ACCEPTING COMMANDS.
014E CD000A              CALL    HOME
0151 CD260A              CALL    RETN  ;INITIALIZE PRINT POSITION
0154 CD9D0A              CALL    PRIMM ;PRINT PROMPT CHARACTER
0157 23                  DB      '#'
0158 00                  DB      00H
                    ;  THREE CHARACTERS WILL NOW BE RECEIVED AND
                    ;  CHECKED TO SEE IF THEY CORRESPOND TO A
                    ;  VALID COMMAND.
0159 1E03                MVI  E, 03D   ;LOAD DIGIT COUNTER
015B CDC701         RDMOR: CALL  CHKEY ;READ KEYBOARD
015E 1D                  DCR     E     ;DECREMENT CHAR. COUNTER
015F C25B01              JNZ     RDMOR ;RECEIVE ANOTHER CHARACTER
                    ;  THREE CHARACTERS HAVE BEEN RECEIVED
                    ;  AND STORED ON THE PRINT STACK AT LOCATIONS
                    ;  CHR2, CHR3, AND CHR4.
                    ;  KYTAB IS A TABLE CONTAINING THE VALID
                    ;  COMMAND CODES.
0162 21E811              LXI  H, KYTAB ;LOAD HL WITH TABLE LOC.
0165 0E32                MVI  C, 50D   ;ALLOW 50 ENTRIES
                    ;  SEARCH TABLE FOR CORRECT ROUTINE
0167 3A0120         FIND1: LDA   CHR2  ;GET FIRST CHARACTER
016A BE                  CMP     M     ;COMPARE WITH TABLE ENTRY
016B CA7A01              JZ      SET2  ;FIRST CHAR. IS CORRECT
016E 23                  INX     H
016F 23                  INX     H
0170 23                  INX     H
0171 23                  INX     H
0172 23                  INX     H     ;POINT HL AT NEXT ENTRY
0173 0D                  DCR     C     ;DECREMENT CHECK COUNTER
0174 C26701              JNZ     FIND1 ;CHECK NEXT TABLE ENTRY
                    ;  THE CODE CANNOT BE FOUND IN THE TABLE,
                    ;  ASSUME THAT THERE HAS BEEN AN ERROR.
0177 C39E09              JMP     ERROR ;ERROR EXIT
                    ;  FIRST CHARACTER IS CORRECT, CHECK SECOND
017A 23             SET2:  INX   H     ;POINT HL AT SECOND CHAR.
017B 0E0A                MVI  C, 10D   ;ALLOW 10 TO BE CHECKED
017D 3A0220         FIND2: LDA   CHR3  ;GET SECOND CHARACTER
0180 BE                  CMP     M     ;COMPARE WITH TABLE ENTRY
0181 CA9001              JZ      TEST1 ;SECOND CHAR. IS CORRECT
0184 23                  INX     H
0185 23                  INX     H
0186 23                  INX     H
```

```
0187 23                 INX     H
0188 23                 INX     H      ;POINT HL AT NEXT ENTRY
0189 0D                 DCR     C      ;DECREMENT CHECK COUNTER
018A C27D01             JNZ     FIND2  ;CHECK NEXT TABLE ENTRY
018D C39E09             JMP     ERROR  ;ERROR EXIT
                ;  SECOND CHARACTER IS VALID,
                ;  CHECK FIRST CHARACTER AGAIN.
0190 2B         TEST1:  DCX     H      ;POINT HL AT FIRST CHAR.
0191 3A0120             LDA     CHR2   ;GET FIRST CHARACTER
0194 BE                 CMP     M      ;COMPARE WITH TABLE ENTRY
0195 C29E09             JNZ     ERROR  ;ERROR EXIT
0198 23                 INX     H
0199 23                 INX     H      ;POINT HL AT THIRD CHAR.
019A 0E0A               MVI  C, 10D    ;ALLOW 10  TO BE CHECKED
019C 3A0320     FIND3:  LDA     CHR4   ;GET THIRD CHARACTER
019F BE                 CMP     M      ;COMPARE WITH TABLE ENTRY
01A0 CAAF01             JZ      TEST2  ;THIRD CHAR. IS CORRECT
01A3 23                 INX     H
01A4 23                 INX     H
01A5 23                 INX     H
01A6 23                 INX     H
01A7 23                 INX     H      ;POINT HL AT NEXT ENTRY
01A8 0D                 DCR     C      ;DECREMENT CHECK COUNTER
01A9 C29C01             JNZ     FIND3  ;CHECK NEXT TABLE ENTRY
01AC C39E09             JMP     ERROR  ;ERROR EXIT
                ;  THIRD CHARACTER IS VALID, CHECK SECOND AGAIN
01AF 2B         TEST2:  DCX     H      ;POINT HL AT SECOND CHAR.
01B0 3A0220             LDA     CHR3   ;GET SECOND CHARACTER
01B3 BE                 CMP     M      ;COMPARE WITH TABLE ENTRY
01B4 C29E09             JNZ     ERROR  ;ERROR EXIT
                ; THIRD AND SECOND CHARACTERS ARE VALID,
                ;  RECHECK THE FIRST CHARACTER.
01B7 2B                 DCX     H      ;POINT HL AT FIRST CHAR.
01B8 3A0120             LDA     CHR2   ;GET FIRST CHARACTER
01BB BE                 CMP     M      ;COMPARE WITH TABLE ENTRY
01BC C29E09             JNZ     ERROR  ;ERROR EXIT
                ;  A VALID COMMAND HAS BEEN RECEIVED.
                ;  JUMP TO APPROPRIATE ROUTINE
                ;  THE ADDRESS OF THE KEYBOARD SERVICE ROUTINE
                ;  IS STORED IN THE TABLE DIRECTLY AFTER THE
                ;  CHARACTER CODE. (SEE TABLE KYTAB)
01BF 23                 INX     H
01C0 23                 INX     H
01C1 23                 INX     H      ;POINT AT HIGH ORDER ADD.
01C2 5E                 MOV  E, M      ;LOAD HIGH ORDER ADDRESS
01C3 23                 INX     H      ;POINT AT LOW ORDER ADD.
01C4 56                 MOV  D, M      ;LOAD LOW ORDER ADDRESS
01C5 EB                 XCHG           ;LOAD HL WITH ADDRESS
01C6 E9                 PCHL           ;GO TO CORRECT SUBROUTINE
```

```
                   ;**********************************************
                   ;*                                            *
                   ;*      SUBROUTINE TO READ THE KEYBOARD       *
                   ;*                                            *
                   ;*   UPON RECEIVING A STROBE FROM THE         *
                   ;* KEYBOARD, THE ASCII DATA IS LOADED INTO    *
                   ;* THE PRINT STACK AND PRINTED.               *
                   ;*   THE DATA IS ALSO RETURNED IN A.          *
                   ;*   IF THE CHARACTER IS A LOWER CASE LETTER  *
                   ;* IT IS CONVERTED TO UPPER CASE.             *
                   ;*   BECAUSE OF THE ERROR EXIT, THIS SUBR.    *
                   ;* CAN ONLY BE CALLED BY THE MAIN PROGRAM     *
                   ;*                                            *
                   ;**********************************************
                   ;THIS ROUTINE WILL WAIT 2 SECONDS FOR A
                   ;KEYBOARD ENTRY. IF NO ENTRY IS RECEIVED BEFORE
                   ;2 SECONDS, THE PRINTHEAD IS MOVED TWO
                   ;POSITIONS TO THE RIGHT SO THE LAST CHARACTER
                   ;PRINTED CAN BE SEEN. SINCE THE TIMER HAS A
                   ;MAXIMUM DELAY OF 1/4 SECOND, 8 DELAY LOOPS ARE
                   ;COUNTED FOR THE 2 SECONDS. REGISTER B CONTAINS
                   ;A FLAG THAT IS SET IF THE PRINTHEAD HAS BEEN
                   ;MOVED TO THE RIGHT.
01C7 C5            CHKEY: PUSH    B      ;STORE BC REGISTERS
01C6 E5                   PUSH    H      ;STORE HL REGISTERS
01C9 0E08                 MVI  C, 08H    ;PUT 8 INTO COUNTER
01CB 0600                 MVI  B, 00H    ;CLEAR FLAG REGISTER
                   ;  SET TIMER FOR 1/4 SECOND
01CD 3E00          LOAD:  MVI  A, 00H
01CF D304                 OUT     STPER ;CLEAR FLIP FLOP
01D1 3E40                 MVI  A, 40H
01D3 D304                 OUT     STPER ;ENABLE FLIP FLOP
01D5 3E78                 MVI  A, 78H
01D7 D300                 OUT     MODEA ;SET COUNTER MODE
01D9 3E24                 MVI  A, 24H
01DB D302                 OUT     TIMA1 ;LOAD LOW ORDER COUNT
01DD 3EF4                 MVI  A, 0F4H
01DF D302                 OUT     TIMA1 ;LOAD HIGH ORDER COUNT
                   ;  CHECK FOR KEYBOARD STROBE
01E1 DB07          CKSTB: IN      KEY   ;READ KEYBOARD
01E3 17                   RAL           ;ROTATE D7 INTO CARRY
01E4 D22002               JNC     STACT ;STROBE IS ACTIVE
                   ;  THERE IS NO KEYBOARD STROBE, IS THE
                   ;  MODE SWITCH STILL ON CONTROL?
  1              +        READ    SETSW ;READ SWITCH INPUTS
01E7 1 D5        +        PUSH    D     ;SAVE DE REGISTERS
01E8 1 1602      +        MVI  D, SETSW  ;STORE SWITCH NUMBER IN D
01EA 1 DB06      +        IN      SWCH  ;READ THE SWITCH INPUTS
01EC 1 1F        +NO:     RAR           ;ROTATE D0 INTO CARRY
01ED 1 15        +        DCR  D        ;DECREMENT COUNTER
01EE 1 C2EC01 +           JNZ     NO    ;NOT CORRECT SWITCH YET
01F1 1 D1        +        POP     D     ;RESTORE DE REGISTERS
01F2 D2FE01               JNC     CONT1
                   ;  MODE SWITCH IS ON SCAN THEREFORE EXIT THE
```

```
                 ;  KEYBOARD ROUTINE
0 1F5 C1                POP      B
0 1F6 33                INX      SP
0 1F7 33                INX      SP     ;RESTORE THE STACK
0 1F8 CD000A            CALL     HOME   ;LINE FEED
0 1FB C32301            JMP      BEGIN  ;EXIT KEYBOARD ROUTINE
                 ;  MODE SWITCH IS ON CONTROL.
                 ;  CHECK IF THE PRINTHEAD HAS BEEN MOVED.
0 1FE 78         CONT1: MOV  A, B       ;TEST FLAG
0 1FF A7                ANA      A      ;SET FLAGS
0200 C2E101             JNZ      CKSTB  ;FLAG IS SET
                 ;  THE PRINTHEAD HAS NOT BEEN MOVED.
                 ;  THEREFORE CHECK THE TIMER.
  1              +      READ     TEST   ;CHECK THE TIMER
0203 1 D5        +      PUSH     D      ;SAVE DE REGISTERS
0204 1 1608      +      MVI  D, TEST    ;STORE SWITCH NUMBER IN D
0206 1 DB06      +      IN       SWCH   ;READ THE SWITCH INPUTS
0208 1 1F        +NO:   RAR             ;ROTATE DO INTO CARRY
0209 1 15        +      DCR  D          ;DECREMENT COUNTER
020A 1 C20802 +         JNZ      NO     ;NOT CORRECT SWITCH YET
020D 1 D1        +      POP      D      ;RESTORE DE REGISTERS
020E DAE101             JC       CKSTB  ;TIMER NOT DONE
                 ;  TIMER IS DONE, DECREMENT COUNTER
0211 0D                 DCR      C      ;DECREMENT COUNTER
0212 C2CD01             JNZ      LOAD   ;TIMING NOT COMPLETE
                 ;  2 SECONDS IS OVER, MOVE THE PRINTHEAD
0215 CDE809             CALL     MOVR   ;MOVE PRINTHEAD RIGHT
0218 CDE809             CALL     MOVR   ;MOVE PRINTHEAD RIGHT
                 ;  SET FLAG INDICATING PRINTHEAD HAS MOVED.
021B 06FF               MVI  B, 0FFH    ;SET FLAGS
021D C3E101             JMP      CKSTB  ;WAIT FOR STROBE
                 ;  A STROBE HAS BEEN RECEIVED FROM THE KEYBOARD
0220 DB07        STACT: IN       KEY    ;READ KEYBOARD
0222 2F                 CMA             ;KEYB. USES NEG LOGIC
0223 E67F               ANI      7FH    ;MASK OFF STROBE INPUT
0225 FE7F               CPI      DELETE ;IS CHARACTER A DELETE
0227 C23002             JNZ      CONT2  ;NOT A DELETE
                 ;  CHARACTER IS A DELETE
                 ;  THEREFORE WAIT FOR A NEW COMMAND.
022A C1                 POP      B
022B 33                 INX      SP
022C 33                 INX      SP     ;RESTORE THE STACK
022D C33901             JMP      KEYB   ;RECEIVE COMMAND AGAIN
                 ;  CHARACTER IS NOT A DELETE, SEE IF IT
                 ;  IS A SMALL LETTER.
0230 4F          CONT2: MOV  C, A       ;STORE CHARACTER IN C
0231 E660               ANI      60H
0233 FE60               CPI      60H    ;IS CHAR. A SMALL LETTER
0235 C23C02             JNZ      CAPTL
                 ;  CHAR. IS A SMALL LETTER, CONVERT TO CAPITAL
0238 79                 MOV  A, C
0239 E65F               ANI      5FH    ;CONVERT TO CAPITAL LETTER
023B 4F                 MOV  C, A       ;STORE IN C
                 ;  CHARACTER IS A CAPITAL LETTER
```

```
                ;  CHECK IF PRINTHEAD HAS BEEN MOVED.
023C 78         CAPTL: MOV  A, B      ;CHECK FLAG
023D A7                ANA     A      ;SET FLAGS
023E CA4702            JZ      RESTR  ;EXIT
                ;  PRINTHEAD MUST BE RESET
0241 CDD009            CALL    MOVL   ;MOVE PRINTHEAD LEFT
0244 CDD009            CALL    MOVL   ;MOVE PRINTHEAD LEFT
                ;  PRINTHEAD IS READY TO PRINT THE CHARACTER
0247 2A1220     RESTR: LHLD    POINT  ;POINT HL AT PRINT STACK
024A 71                MOV  M, C      ;LOAD CHAR ON PRINT STACK
024B CD320A            CALL    PRINT  ;PRINT THE CHARACTER
                ;  CHARACTER HAS BEEN PRINTED, MAKE SURE THAT
                ;  THE KEYBOARD STROBE HAS GONE AWAY.
024E DB07       WAIT:  IN      KEY    ;READ KEYBOARD
0250 17                RAL
0251 D24E02            JNC     WAIT   ;WAIT FOR STROBE TO GO
                ;  STROBE HAS GONE AWAY, LOAD CHARACTER SO THAT
                ;  IT CAN BE READ BY CALLING PROGRAM.
0254 79                MOV  A, C      ;PUT DATA INTO A
0255 E1                POP     H      ;RESTORE HL REGISTERS
0256 C1                POP     B      ;RESTORE BC REGISTERS
0257 C9                RET
                ;*********************************************
                ;*                                           *
                ;*      ROUTINE TO DISPLAY MEMORY LOCATIONS  *
                ;*                                           *
                ;*********************************************
                ;THIS KEYBOARD SERVICE ROUTINE WILL ALLOW THE
                ;CONTENTS OF MEMORY TO BE DISPLAYED. SUBROUTINE
                ;GTADD IS USED TO RECEIVE TWO HEXADECIMAL
                ;ADDRESSES FROM THE KEYBOARD.  THE FIRST IS THE
                ;STARTING MEMORY ADDRESS AND THE SECOND IS THE
                ;ENDING MEMORY ADDRESS.
                ;THE PROGRAM READS EACH MEMORY LOCATION AND
                ;PRINTS THE CONTENT OF EACH LOCATION.
0258 CDC701     DPM:   CALL    CHKEY  ;READ KEYBOARD
025B FE20              CPI     ' '    ;IS CHARACTER A SPACE
025D C29E09            JNZ     ERROR  ;NOT A SPACE
                ;  A SPACE HAS BEEN RECEIVED AFTER THE COMMAND.
                ;  READ STARTING ADDRESS
0260 CD4E09            CALL    GTADD  ;GET STARTING ADDRESS
0263 DA9E09            JC      ERROR  ;ERROR EXIT
0266 EB                XCHG           ;EXCHANGE HL AND DE
                ;  WAIT FOR ANOTHER SPACE.
0267 CDC701            CALL    CHKEY  ;READ KEYBOARD
026A FE20              CPI     ' '    ;IS CHARACTER A SPACE
026C CA7F02            JZ      GTSPC  ;CHARACTER IS A SPACE
                ;  A SPACE HAS NOT BEEN RECEIVED, THEREFORE
                ;  CHECK FOR A PERIOD.
026F FE2E              CPI     '.'    ;IS CHARACTER A PERIOD
0271 C29E09            JNZ     ERROR  ;ERROR EXIT
                ;  A PERIOD HAS BEEN RECEIVED, THEREFORE ONLY
                ;  PRINT THE STARTING ADDRESS DATA.
0274 EB                XCHG           ;EXCHANGE HL AND DE
```

```
0275 1E01                MVI  E, 01H
0277 1600                MVI  D, 00H   ;LOAD 1 INTO DE
0279 CD000A              CALL    HOME  ;INITIALIZE PRINTER
027C C39402              JMP     DPMPR ;PRINT RESULTS
                 ;  RECEIVE ENDING ADDRESS.
027F CD4E09      GTSPC:  CALL    GTADD ;GET ENDING ADDRESS
0282 DA9E09              JC      ERROR ;ERROR EXIT
0285 D5                  PUSH    D     ;STORE STARTING ADDRESS
                 ;  SUBTRACT STARTING ADDRESS FROM ENDING ADD.
0286 7A                  MOV  A, D
0287 2F                  CMA
0288 57                  MOV  D, A     ;COMPLEMENT D
0289 7B                  MOV  A, E
028A 2F                  CMA
028B 5F                  MOV  E, A     ;COMPLEMENT E
028C 13                  INX     D
028D 13                  INX     D     ;INCREMENT DE TWICE
028E 19                  DAD     D     ;ADD DE TO HL
028F EB                  XCHG          ;EXCHANGE DE AND HL
0290 E1                  POP     H     ;RESTORE STARTING ADDRESS
0291 CD000A              CALL    HOME  ;INITIALIZE PRINTER
                 ;  DE CONTAINS THE NUMBER OF ADDRESSES TO BE
                 ;  PRINTED. HL CONTAINS THE STARTING ADDRESS.
0294 CDB70A      DPMPR:  CALL    CLPSK ;CLEAR THE PRINT STACK
0297 CD2109              CALL    HLSTK ;PUT ADD. IN PRINT STACK
029A 3E20                MVI  A, ' '
029C 320420              STA     CHR5  ;PUT SPACE IN CHR5
029F 7E                  MOV  A, M     ;READ DATA AT HL
02A0 E6F0                ANI     0F0H  ;MASK OFF LOW NIBBLE
02A2 0F                  RRC
02A3 0F                  RRC
02A4 0F                  RRC
02A5 0F                  RRC           ;MOVE TO LOW NIBBLE
02A6 CDBC08              CALL    CONAS ;CONVERT TO ASCII
02A9 DA9E09              JC      ERROR ;ERROR EXIT
                 ;  PUT MS NIBBLE INTO PRINT STACK.
02AC 320520              STA     CHR6  ;STORE IN CHR6
02AF 7E                  MOV  A, M     ;READ DATA AT HL
02B0 E60F                ANI     0FH   ;MASK OFF HIGH NIBBLE
02B2 CDBC08              CALL    CONAS ;CONVERT TO ASCII
02B5 DA9E09              JC      ERROR ;ERROR EXIT
                 ;  PUT LS NIBBLE INTO PRINT STACK.
02B8 320620              STA     CHR7  ;STORE IN CHR7
02BB CD860A              CALL    PSTAK ;PRINT ENTIRE STACK
                 ;  POINT HL AT NEXT LOCATION AND
                 ;  DECREMENT COUNTER.
02BE 23                  INX     H     ;INCREMENT HL
02BF 1B                  DCX     D     ;DECREMENT COUNTER
                 ;  CHECK THE COUNTER
02C0 7B                  MOV  A, E     ;GET E REGISTER
02C1 A7                  ANA     A     ;SET FLAGS
02C2 C29402              JNZ     DPMPR ;PRINT ANOTHER LOCATION
02C5 7A                  MOV  A, D     ;GET D REGISTER
02C6 A7                  ANA     A     ;SET FLAGS
```

```
02C7 C29402          JNZ     DPMPR ;PRINT ANOTHER LOCATION
               ;  ALL LOCATIONS HAVE BEEN PRINTED.
02CA C33901          JMP     KEYB
               ;************************************************
               ;*                                              *
               ;*        ROUTINE TO SET MEMORY LOCATIONS       *
               ;*                                              *
               ;************************************************
               ;THIS KEYBOARD SERVICE ROUTINE ALLOWS THE USER
               ;TO SET MEMORY LOCATIONS. SUBROUTINE GTADD IS
               ;USED TO RECEIVE, FROM THE KEYBOARD, THE MEMORY
               ;ADDRESS TO BE SET. THE ROUTINE THEN ACCEPTS
               ;THE DATA FROM THE KEYBOARD AND LOADS IT INTO
               ;THE MEMORY LOCATION.
02CD CDC701    STM:  CALL    CHKEY ;READ THE KEYBOARD
02D0 FE20            CPI     ' '   ;IS CHARACTER A SPACE
02D2 C29E09          JNZ     ERROR ;ERROR EXIT
02D5 CD4E09          CALL    GTADD ;RECEIVE ADDRESS
02D8 CD000A          CALL    HOME  ;LINE FEED
02DB CD260A    STMPR: CALL   RETN  ;CARRIAGE RETURN
               ;  PRINT THE ADDRESS TO BE SET.
02DE CD2109          CALL    HLSTK ;LOAD ADDRESS INTO STACK
02E1 CD320A          CALL    PRINT
02E4 CD320A          CALL    PRINT
02E7 CD320A          CALL    PRINT
02EA CD320A          CALL    PRINT ;PRINT ADDRESS
               ;  ADDRESS HAS BEEN PRINTED.
02ED CDE809          CALL    MOVR  ;GENERATE A SPACE
               ;  WHEN A PERIOD IS RECEIVED INSTEAD OF DATA,
               ; EXIT FROM THE ROUTINE.
02F0 CDC701          CALL    CHKEY ;READ KEYBOARD
02F3 FE2E            CPI     '.'   ;IS CHARACTER A PERIOD
02F5 CA3901          JZ      KEYB  ;ROUTINE IS FINISHED
               ;  SINCE DATA FROM THE KEYBOARD IS IN ASCII
               ;  IT MUST BE CONVERTED TO HEX.
02F8 CD9D08          CALL    CONHX ;CONVERT TO HEX
02FB DA9E09          JC      ERROR ;ERROR EXIT
02FE 07              RLC
02FF 07              RLC
0300 07              RLC
0301 07              RLC           ;MOVE INTO HIGH NIBBLE
0302 47              MOV  B, A     ;STORE IN REG B
0303 CDC701          CALL    CHKEY ;READ KEYBOARD
0306 CD9D08          CALL    CONHX ;CONVERT TO HEX
0309 DA9E09          JC      ERROR ;ERROR EXIT
               ;  A CONTAINS THE LS NIBBLE
               ;  B CONTAINS THE MS NIBBLE OF THE DATA
030C B0              ORA     B     ;OR WITH REG B
030D 77              MOV  M, A     ;LOAD DATA
030E 23              INX     H     ;INCREMENT HL
030F CD000A          CALL    HOME  ;LINE FEED
0312 C3DB02          JMP     STMPR ;PRINT NEXT ADDRESS
```

```
                ;*********************************************
                ;*                                           *
                ;*          ROUTINE TO RUN PROGRAM           *
                ;*                                           *
                ;*********************************************
                ;THIS KEYBOARD SERVICE ROUTINE ALLOWS THE USER
                ; TO INITIATE PROGRAM EXCECUTION AT ANY DESIRED
                ;ADDRESS.  GTADD IS USED TO RECEIVE THE ADDRESS
                ;UPON RECEIPT OF A PERIOD, THE PROGRAM WILL
                ;BEGIN EXECUTION FROM THE LOADED ADDRESS.
0315 CDC701     RUN:   CALL    CHKEY ;READ KEYBOARD
0318 FE20              CPI     ' '   ;IS CHARACTER A SPACE
031A C29E09            JNZ     ERROR ;ERROR EXIT
031D CD4E09            CALL    GTADD ;GET STARTING ADDRESS
                ;  ADDRESS HAS BEEN RECEIVED, WAIT FOR A PERIOD
0320 CDC701            CALL    CHKEY ;READ KEYBOARD
0323 FE2E              CPI     '.'   ;IS CHARACTER A PERIOD
0325 C29E09            JNZ     ERROR ;ERROR EXIT
                ;  MOVE PRINTER TO THE HOME POSITION,
                ;  AND RUN THE PROGRAM.
0328 CD000A            CALL    HOME  ;GENERATE LINE FEED
032B E9                PCHL          ;RUN AT ADDRESS
                ;*********************************************
                ;*                                           *
                ;*        ROUTINE TO READ INPUT PORT         *
                ;*                                           *
                ;*********************************************
                ;THIS KEYBOARD SERVICE ROUTINE ALLOWS THE USER
                ;TO READ AN INPUT PORT.  THE PORT ADDRESS IS
                ;ENTERED FIRST AND THE CONTENTS ARE THEN
                ;PRINTED.
032C CDC701     INP:   CALL    CHKEY ;READ KEYBOARD
032F FE20              CPI     ' '   ;IS CHARACTER A SPACE
0331 C29E09            JNZ     ERROR ;ERROR EXIT
                ;  RECEIVE MS NIBBLE OF PORT ADDRESS
0334 CDC701            CALL    CHKEY ;READ KEYBOARD
0337 CD9D08            CALL    CONHX ;CONVERT TO HEX
033A DA9E09            JC      ERROR ;ERROR EXIT
033D 07                RLC
033E 07                RLC
033F 07                RLC
0340 07                RLC           ;ROTATE INTO HIGH NIBBLE
0341 47                MOV  B, A     ;STORE IN REG B
                ;  RECEIVE LS NIBBLE OF PORT ADDRESS
0342 CDC701            CALL    CHKEY ;READ KEYBOARD
0345 CD9D08            CALL    CONHX ;CONVERT TO HEX
0348 DA9E09            JC      ERROR ;ERROR EXIT
                ;  A NOW CONTAINS THE LS NIBBLE AND
                ;  B CONTAINS THE MS NIBBLE OF THE PORT ADDRESS
034B B0                ORA     B     ;LOAD LOW NIBBLE
034C 321020            STA     CHR17 ;STORE AT CHR17
                ;  THE INSTRUCTION (IN PORT) IS STORED
                ;  IN RAM MEMORY FOLLOWED BY A RETURN
034F 3EDB              MVI  A, 0DBH  ;LOAD OPCODE
```

```
0351 320F20           STA     CHR16 ;STORE AT CHR16
0354 3EC9             MVI  A, 0C9H  ;LOAD RET
0356 321120           STA     CHR18 ;STORE AT CHR18
                ;  THE SUBROUTINE IS NOW CALLED
0359 CD0F20           CALL    CHR16 ;CALL INPUT ROUTINE
                ;  A NOW CONTAINS THE PORT DATA IN HEXADECIMAL
                ;  IT MUST NOW BE CONVERTED TO TWO
                ;  ASCII CHARACTERS AND PRINTED.
035C 47               MOV  B, A     ;STORE INPUT IN REG B
035D E6F0             ANI     0F0H  ;MASK OFF LOW NIBBLE
035F 0F               RRC
0360 0F               RRC
0361 0F               RRC
0362 0F               RRC           ;ROTATE INTO LOW NIBBLE
                ;  MS NIBBLE IS NOW CONVERTED TO ASCII
0363 CDBC08           CALL    CONAS ;CONVERT TO ASCII
0366 320820           STA     CHR9  ;STORE AT CHR9
0369 3E3D             MVI  A, '='   ;LOAD EQUAL SIGN
036B 320720           STA     CHR8  ;STORE AT CHR8
036E CD320A           CALL    PRINT ;PRINT =
0371 CD320A           CALL    PRINT ;PRINT HIGH NIBBLE
                ;  EQUAL SIGN AND MS NIBBLE HAVE BEEN PRINTED
0374 78               MOV  A, B     ;GET DATA
0375 E60F             ANI     0FH   ;MASK OFF HIGH NIBBLE
0377 CDBC08           CALL    CONAS ;CONVERT TO ASCII
037A 320920           STA     CHR10 ;STORE AT CHR10
037D CD320A           CALL    PRINT ;PRINT LOW NIBBLE
                ;  DATA HAS BEEN PRINTED
0380 C33901           JMP     KEYB  ;RETURN FROM ROUTINE
                ;*********************************************
                ;*                                           *
                ;*       ROUTINE TO OUTPUT DATA TO PORT      *
                ;*                                           *
                ;*********************************************
                ;THIS KEYBOARD SERVICE ROUTINE ALLOWS THE USER
                ;TO OUTPUT DATA TO A PORT.
0383 CDC701     OUTP: CALL    CHKEY ;READ KEYBOARD
0386 FE20             CPI     ' '   ;IS CHARACTER A SPACE
0388 C29E09           JNZ     ERROR ;ERROR EXIT
                ;  RECEIVE MS NIBBLE OF PORT ADDRESS
038B CDC701           CALL    CHKEY ;READ KEYBOARD
038E CD9D08           CALL    CONHX ;CONVERT TO HEX
0391 DA9E09           JC      ERROR ;ERROR EXIT
0394 07               RLC
0395 07               RLC
0396 07               RLC
0397 07               RLC           ;ROTATE INTO UPPER NIBBLE
0398 47               MOV  B, A     ;STORE IN REG B
                ;  RECEIVE LS NIBBLE OF PORT ADDRESS
0399 CDC701           CALL    CHKEY ;READ KEYBOARD
039C CD9D08           CALL    CONHX ;CONVERT TO HEX
039F DA9E09           JC      ERROR ;ERROR EXIT
                ;  A NOW CONTAINS THE LS NIBBLE AND
                ;  B CONTAINS THE MS NIBBLE OF THE PORT ADDRESS
```

```
03A2 B0                 ORA     B     ;OR WITH B
                  ;  LOAD PROGRAM (OUT PORT) INTO RAM MEMORY
03A3 321020             STA     CHR17 ;STORE AT CHR17
03A6 3ED3               MVI  A, 0D3H  ;LOAD OPCODE
03A8 320F20             STA     CHR16 ;LOAD INTO CHR16
                  ;  LOAD RETURN STATEMENT INTO MEMORY
03AB 3EC9               MVI  A, 0C9H  ;LOAD RET
03AD 321120             STA     CHR18 ;LOAD INTO CHR18
                  ;  WAIT FOR A SPACE
03B0 CDC701             CALL    CHKEY ;READ KEYBOARD
03B3 FE20               CPI     ' '   ;IS CHARACTER A SPACE
03B5 C29E09             JNZ     ERROR ;ERROR EXIT
                  ;  READ MS NIBBLE OF PORT DATA
03B8 CDC701             CALL    CHKEY ;READ KEYBOARD
03BB CD9D08             CALL    CONHX ;CONVERT TO HEX
03BE DA9E09             JC      ERROR ;ERROR EXIT
03C1 07                 RLC
03C2 07                 RLC
03C3 07                 RLC
03C4 07                 RLC           ;ROTATE INTO HIGH NIBBLE
03C5 47                 MOV  B, A     ;STORE IN REG B
                  ;  READ LS NIBBLE OF PORT DATA
03C6 CDC701             CALL    CHKEY ;READ KEYBOARD
03C9 CD9D08             CALL    CONHX ;CONVERT TO HEX
03CC DA9E09             JC      ERROR ;ERROR EXIT
                  ;  A NOW CONTAINS THE LS NIBBLE AND
                  ;  B CONTAINS THE MS NIBBLE OF THE PORT DATA
03CF B0                 ORA     B     ;PUT IN LOW NIBBLE
                  ;  CALL THE PROGRAM WHICH WAS STORED IN MEMORY.
                  ;  THE DATA IS THEN OUTPUT TO THE PORT
03D0 CD0F20             CALL    CHR16 ;CALL OUTPUT ROUTINE
03D3 C33901             JMP     KEYB  ;RETURN FROM ROUTINE
                  ;*********************************************
                  ;*                                           *
                  ;*       ROUTINE TO DISPLAY PARAMETERS       *
                  ;*                                           *
                  ;*********************************************
                  ;THIS KEYBOARD SERVICE ROUTINE ALLOWS THE USER
                  ;TO DISPLAY VARIOUS PARAMETERS IN THE SYSTEM.
                  ;THE PROGRAM ITSELF IS VERY SIMILIAR TO THE
                  ;KEYBOARD ROUTINE (KEYB). IT ACCEPTS THREE
                  ;CHARACTERS AND THEN CHECKS TO SEE IF THE
                  ;CHARACTER CODE CORRESPONDS TO A VALID
                  ;CODE CONTAINED IN THE TABLE (DISTB).
03D6 CDC701       DSP:  CALL    CHKEY ;READ KEYBOARD
03D9 FE20               CPI     ' '   ;IS CHAR. A SPACE
03DB C29E09             JNZ     ERROR ;NOT A SPACE
                  ;  THE THREE CHARACTERS WILL NOW BE RECEIVED
                  ;  AND CHECKED TO SEE IF THEY CORRESPOND TO
                  ;  A VALID PARAMETER NAME.
03DE 1E03               MVI  E, 03H   ;LOAD DIGIT COUNTER
03E0 CDC701       RDMR1: CALL   CHKEY ;READ KEYBOARD
03E3 1D                 DCR     E     ;DECREMENT CHAR. COUNTER
03E4 C2E003             JNZ     RDMR1 ;RECEIVE ANOTHER CHAR.
```

```
                ;  THREE CHARACTERS HAVE BEEN RECEIVED
                ;  AND STORED IN THE PRINT STACK AT LOCATIONS
                ;  CHR6, CHR7, AND CHR8.
                ;  DISTB IS A TABLE CONTAINING THE VALID
                ;  PARAMETER NAMES.
03E7 219704            LXI  H, DISTB ;LOAD HL WITH TABLE LOC.
03EA 0E14              MVI  C, 20D   ;ALLOW 20  TO BE CHECKED
                ;  SEARCH TABLE FOR CORRECT ROUTINE
03EC 3A0520     FND1:  LDA      CHR6  ;GET FIRST CHAR.
03EF BE                CMP      M     ;COMPARE WITH TABLE ENTRY
03F0 CAFF03            JZ       ST2   ;FIRST CHAR. IS CORRECT
03F3 23                INX      H
03F4 23                INX      H
03F5 23                INX      H
03F6 23                INX      H
03F7 23                INX      H     ;POINT HL AT NEXT ENTRY
03F8 0D                DCR      C     ;DECREMENT CHECK COUNTER
03F9 C2EC03            JNZ      FND1  ;CHECK NEXT TABLE ENTRY
03FC C39E09            JMP      ERROR ;ERROR EXIT
                ;  FIRST CHARACTER IS CORRECT,
                ;  CHECK SECOND CHARACTER.
03FF 23         ST2 :  INX      H     ;POINT HL AT SECOND CHAR.
0400 0E0A              MVI  C,  10D   ;ALLOW 20 TO BE CHECKED
0402 3A0620     FND2:  LDA      CHR7  ;GET SECOND CHAR.
0405 BE                CMP      M     ;COMPARE WITH TABLE ENTRY
0406 CA1504            JZ       TST1  ;SECOND CHAR. IS CORRECT
0409 23                INX      H
040A 23                INX      H
040B 23                INX      H
040C 23                INX      H
040D 23                INX      H     ;POINT HL AT NEXT ENTRY
040E 0D                DCR      C     ;DECREMENT CHECK COUNTER
040F C20204            JNZ      FND2  ;CHECK NEXT TABLE ENTRY
0412 C39E09            JMP      ERROR ;ERROR EXIT
                ;  SECOND CHARACTER IS VALID,
                ;  RECHECK FIRST CHARACTER.
0415 2B         TST1:  DCX      H     ;POINT HL AT FIRST CHAR.
0416 3A0520            LDA      CHR6  ;GET FIRST CHAR.
0419 BE                CMP      M     ;COMPARE WITH TABLE ENTRY
041A C29E09            JNZ      ERROR ;ERROR EXIT
041D 23                INX      H
041E 23                INX      H     ;POINT HL AT THIRD CHAR.
041F 0E0A              MVI  C,  10D   ;ALLOW 10 TO BE CHECKED
0421 3A0720     FND3:  LDA      CHR8  ;GET THIRD CHAR.
0424 BE                CMP      M     ;COMPARE WITH TABLE ENTRY
0425 CA3404            JZ       TST2  ;THIRD CHAR. IS CORRECT
0428 23                INX      H
0429 23                INX      H
042A 23                INX      H
042B 23                INX      H
042C 23                INX      H     ;POINT HL AT NEXT ENTRY
042D 0D                DCR      C     ;DECREMENT CHECK COUNTER
042E C22104            JNZ      FND3  ;CHECK NEXT TABLE ENTRY
0431 C39E09            JMP      ERROR ;ERROR EXIT
```

```
                ;  THIRD CHARACTER IS CORRECT,
                ;  CHECK SECOND CHARACTER.
0434 2B         TST2:  DCX     H      ;POINT HL AT SECOND CHAR.
0435 3A0620            LDA     CHR7   ;GET SECOND CHAR.
0438 BE                CMP     M      ;COMPARE WITH TABLE ENTRY
0439 C29E09            JNZ     ERROR  ;ERROR EXIT
                ;  THIRD AND SECOND CHARACTERS ARE VALID,
                ;  RECHECK FIRST CHARACTER.
043C 2B                DCX     H      ;POINT HL AT FIRST CHAR.
043D 3A0520            LDA     CHR6   ;GET FIRST CHAR.
0440 BE                CMP     M      ;COMPARE WITH TABLE ENTRY
0441 C29E09            JNZ     ERROR  ;ERROR EXIT
                ;  A VALID PARAMETER NAME HAS BEEN RECEIVED
                ;  JUMP TO APPROPRIATE ROUTINE
                ;  THE ADDRESSES OF THE DISPLAY ROUTINES ARE
                ;  STORED IN THE TABLE DIRECTLY AFTER THE
                ;  CHARACTER CODE (SEE TABLE DISTB).
0444 23                INX     H
0445 23                INX     H
0446 23                INX     H        ;POINT AT HIGH ORDER ADD.
0447 5E                MOV  E, M        ;LOAD HIGH ORDER ADDRESS
0448 23                INX     H        ;POINT AT LOW ORDER ADD.
0449 56                MOV  D, M        ;LOAD LOW ORDER ADDRESS
044A EB                XCHG             ;LOAD HL WITH ADDRESS
044B E9                PCHL             ;GO TO CORRECT SUBROUTINE
                ;
                ;SUBROUTINE TO PRINT THE DATA IN THE HL REG
                ;  THE NUMBER IS PRINTED IN HEXADICIMAL FORM
044C CD9D0A     PHL:   CALL    PRIMM ;PRINT EQUAL SIGN
044F 3D20              DB      '= '
0451 00                DB      00H
                ;  PRINT MS NIBBLE OF MS BYTE
0452 7C                MOV  A, H      ;GET H REGISTER
0453 E6F0              ANI     0F0H   ;MASK OFF LOW NIBBLE
0455 0F                RRC
0456 0F                RRC
0457 0F                RRC
0458 0F                RRC            ;ROTATE INTO LOW NIBBLE
0459 CDBC08            CALL    CONAS  ;CONVERT TO ASCII
045C E5                PUSH H
045D 2A1220            LHLD    POINT
0460 77                MOV  M, A
0461 E1                POP  H
0462 CD320A            CALL    PRINT  ;PRINT THE CHARACTER
                ;  PRINT LS NIBBLE OF MS BYTE
0465 7C                MOV  A, H      ;GET H REGISTER
0466 E60F              ANI     0FH    ;MASK OFF HIGH NIBBLE
0468 CDBC08            CALL    CONAS  ;CONVERT TO ASCII
046B E5                PUSH H
046C 2A1220            LHLD    POINT
046F 77                MOV  M, A
0470 E1                POP  H
0471 CD320A            CALL    PRINT  ;PRINT THE CHARACTER
                ;  PRINT MS NIBBLE OF LS BYTE
```

```
0474 7D              MOV  A, L        ;GET L REGISTER
0475 E6F0            ANI       0F0H   ;MASK OFF LOW NIBBLE
0477 0F              RRC
0478 0F              RRC
0479 0F              RRC
047A 0F              RRC              ;ROTATE INTO LOW NIBBLE
047B CDBC08          CALL      CONAS  ;CONVERT TO ASCII
047E E5              PUSH H
047F 2A1220          LHLD      POINT
0482 77              MOV  M, A
0483 E1              POP  H
0484 CD320A          CALL      PRINT  ;PRINT THE CHARACTER
              ;  PRINT LS NIBBLE OF LS BYTE
0487 7D              MOV  A, L        ;GET L REGISTER
0488 E60F            ANI       0FH    ;MASK OFF HIGH NIBBLE
048A CDBC08          CALL      CONAS  ;CONVERT TO ASCII
048D E5              PUSH H
048E 2A1220          LHLD      POINT
0491 77              MOV  M, A
0492 E1              POP  H
0493 CD320A          CALL      PRINT  ;PRINT THE CHARACTER
              ;  THE ENTIRE NUMBER HAS BEEN PRINTED
0496 C9              RET
              ;
              ;JUMP TABLE FOR DISPLAY ROUTINE
0497 414C4C   DISTB: DB        'ALL'
049A E204            DW        ALL
049C 43454E          DB        'CEN'
049F A306            DW        DCEN
04A1 43484E          DB        'CHN'
04A4 AD07            DW        DCHN
04A6 435349          DB        'CSI'
04A9 F206            DW        DCSI
04AB 435354          DB        'CST'
04AE AC06            DW        DCST
04B0 435450          DB        'CTP'
04B3 0507            DW        DCTP
04B5 444953          DB        'DIS'
04B8 1807            DW        DDIS
04BA 464C47          DB        'FLG'
04BD 1708            DW        DFLG
04BF 4D4C43          DB        'MLC'
04C2 B506            DW        DMLC
04C4 504354          DB        'PCT'
04C7 C806            DW        DPCT
04C9 504F53          DB        'POS'
04CC FB07            DW        DPOS
04CE 505249          DB        'PRI'
04D1 CB07            DW        DPRI
04D3 534143          DB        'SAC'
04D6 3E07            DW        DSAC
04D8 534349          DB        'SCI'
04DB EC07            DW        DSCI
04DD 54414E          DB        'TAN'
```

```
04E0 DF06            DW      DTAN
                ;
                ;SUBROUTINE TO DISPLAY PARAMETERS WHICH HAVE
                ;BEEN CHANGED FROM THEIR DEFAULT VALUES.
                ;EACH PARAMETER IS FIRST COMPARED TO ITS
                ;DEFAULT VALUE. IF THE PARAMTER IS STILL SET TO
                ;ITS DEFAULT VALUE, IT IS NOT PRINTED.
04E2 CD000A     ALL:    CALL    HOME  ;LINE FEED
04E5 CD260A             CALL    RETN  ;CARRIAGE RETURN
04E8 CD9D0A             CALL    PRIMM
04EB 2054494D           DB      ' TIM'
04EF 00                 DB      00H
04F0 CDA50E             CALL    TIME  ;PRINT THE TIME
                ;  CHECK SCAN INTERVAL
04F3 3A3322             LDA     SCAN1
04F6 FE01               CPI     1
04F8 CA1505             JZ      DPS   ;DEFAULT VALUE
                ;  SCAN INTERVAL CHANGED FROM DEFAULT
04FB CD000A             CALL    HOME  ;LINE FEED
04FE CD260A             CALL    RETN  ;CARRIAGE RETURN
0501 CD9D0A             CALL    PRIMM
0504 5343414E           DB   'SCAN INT= '
0508 20494E54
050C 3D20
050E 00                 DB   00H
050F 3A3322             LDA     SCAN1
0512 CD2108             CALL    PRBYT ;PRINT SCAN INTERVAL
                ;  DISPLAY PRINTER STATUS
0515 3A2F22     DPS:    LDA     PROF  ;READ PRINTER FLAG
0518 A7                 ANA  A        ;SET FLAGS
0519 CA5505             JZ      CKCSI ;DEFAULT
                ;  PRINTER HAS BEEN ENABLED
051C CD000A             CALL    HOME  ;LINE FEED
051F CD260A             CALL    RETN  ;CARRIAGE RETURN
0522 CD9D0A             CALL    PRIMM
0525 5052494E           DB   'PRINT INT= '
0529 5420494E
052D 543D20
0530 00                 DB   00H
0531 3A3122             LDA     PINT
0534 CD2108             CALL    PRBYT ;PRINT PRINT INTERVAL
                ;  DISPLAY ACTIVITY PRINT FLAG
0537 3A1A21             LDA     ACLOC
053A A7                 ANA  A        ;SET FLAGS
053B CA5505             JZ      CKCSI ;DEFAULT
                ;  ACTIVITIES WILL BE PRINTED
053E CD000A             CALL    HOME  ;LINE FEED
0541 CD260A             CALL    RETN  ;CARRIAGE RETURN
0544 CD9D0A             CALL    PRIMM
0547 464C4147           DB   'FLAGS PRINTED'
054B 53205052
054F 494E5445
0553 44
0554 00                 DB      00H
```

```
                   ;  DISPLAY THE CASSETTE STORE INTERVAL
0555 3A3622   CKCSI: LDA     CASI1 ;LOAD CASSETTE INTERVAL
0558 FE01            CPI     1     ;CCMP. WITH DEFAULT VALUE
055A CA7805          JZ      CKDIS ;DEFAULT VALUE
              ;  CASSETTE STORE INTERVAL CHANGED
055D CD000A          CALL    HOME  ;LINE FEED
0560 CD260A          CALL    RETN  ;CARR RETURN
0563 CD9D0A          CALL    PRIMM
0566 43415353        DB   'CASS. INTERVAL'
056A 2E20494E
056E 54455256
0572 414C
0574 00              DB      00H
0575 CDF806          CALL    PRCSI
              ;  DISPLAY THE DISTANCE AT WHICH SCANNING
              ;  RATE WILL INCREASE
0578 3A1C21   CKDIS: LDA     DISTC ;LOAD VALUE
057B FE00            CPI     0     ;COMPARE TO DEFAULT
057D CA9A05          JZ      CKCN
              ; RATE INCREASE DISTANCE HAS BEEN CHANGED
0580 CD000A          CALL    HOME  ;LINE FEED
0583 CD260A          CALL    RETN  ;CARRIAGE RETURN
0586 CD9D0A          CALL    PRIMM
0589 52415445        DB   'RATE INC DIST'
058D 20494E43
0591 20444953
0595 54
0596 00              DB      00H
0597 CD3107          CALL    PRDIS
              ;  DISPLAY THE CHANNEL
059A 3A3520   CKCN:  LDA     CHNL  ;GET CHANNEL NUMBER
059D A7              ANA  A        ;SET FLAGS
059E CAB905          JZ      CKCH  ;NC CHANNEL SELECTED
              ;  A CHANNEL HAS BEEN SELECTED
05A1 CD000A          CALL    HOME  ;LINE FEED
05A4 CD260A          CALL    RETN  ;CARRIAGE RETURN
05A7 CD9D0A          CALL    PRIMM
05AA 4348414E        DB      'CHANNEL '
05AE 4E454C20
05B2 00              DB      00H
05B3 3A3520          LDA     CHNL  ;GET CHANNEL NUMBER
05B6 CD2108          CALL    PRBYT ;PRINT CHANNEL NUMBER
              ;  DISPLAY ACTIVE CHANNELS
05B9 3A3620   CKCH:  LDA     CHN1
05BC A7              ANA  A        ;SET FLAGS
05BD CAC305          JZ      CKCEN
              ;  ACTIVE CHANNELS HAVE BEEN SET,
              ;  PRINT THE ACTIVE CHANNEL NUMBERS
05C0 CD6A07          CALL    DSAC2
              ;  CHECK CEN
05C3 2A1F20   CKCEN: LHLD    CEND  ;LCAD CEN
05C6 7C              MOV  A, H     ;CHECK MSB
05C7 FE3F            CPI     3FH
05C9 C2D505          JNZ     NOCEN ;CHANGED FROM DEFAULT
```

```
05CC 7D                  MOV  A, L     ;CHECK LSB
05CD FEFF                CPI     0FFH
05CF C2D505              JNZ     NOCEN ;CHANGED FROM DEFAULT
05D2 C3EE05              JMP     CKCST
                ;  CEN HAS BEEN CHANGED
05D5 CD000A     NOCEN:   CALL    HOME  ;LINE FEED
05D8 CD260A              CALL    RETN  ;CARRIAGE RETURN
05DB CD9D0A              CALL    PRIMM
05DE 42554646            DB      'BUFF. END'
05E2 2E20454E
05E6 44
05E7 00                  DB      00H
05E8 2A1F20              LHLD    CEND
05EB CD4C04              CALL    PHL   ;PRINT CEN
                ;  CHECK CST
05EE 2A1D20     CKCST:   LHLD    CSTRT ;LOAD CST
05F1 7C                  MOV  A, H     ;CHECK MSB
05F2 FE30                CPI     30H
05F4 C20006              JNZ     NOCST ;CHANGED FROM DEFAULT
05F7 7D                  MOV  A, L     ;CHECK LSB
05F8 FE00                CPI     00H
05FA C20006              JNZ     NOCST ;CHANGED FROM DEFAULT
05FD C31B06              JMP     CKCTP ;CHECK NEXT PARAMETER
                ;  CST HAS BEEN CHANGED
0600 CD000A     NOCST:   CALL    HOME  ;LINE FEED
0603 CD260A              CALL    RETN  ;CARRIAGE RETURN
0606 CD9D0A              CALL    PRIMM
0609 42554646            DB      'BUFF. START'
060D 2E205354
0611 415254
0614 00                  DB      00H
0615 2A1D20              LHLD    CSTRT
0618 CD4C04              CALL    PHL   ;PRINT CST
                ;  CHECK CTP
061B 3A2F20     CKCTP:   LDA     TPLEN ;LOAD TPLEN
061E FE1E                CPI     30D
0620 CA3B06              JZ      CKMLT ;DEFAULT VALUE
                ;  CTP HAS BEEN CHANGED
0623 CD000A              CALL    HOME  ;LINE FEED
0626 CD260A              CALL    RETN  ;CARRIAGE RETURN
0629 CD9D0A              CALL    PRIMM
062C 54415045            DB      'TAPE LENGTH'
0630 204C454E
0634 475448
0637 00                  DB      00H
0638 CD0B07              CALL    PRCTP ;PRINT CTP
                ;  CHECK MULTIPLIER CONSTANT
063B 3A3F22     CKMLT:   LDA     CMULT ;LOAD CMULT
063E FE32                CPI     50    ;COMPARE WITH DEFAULT
0640 CA5E06              JZ      CKTAN ;DEFAULT VALUE
                ;  MULTIPLIER CONSTANT HAS BEEN CHANGED
0643 CD000A              CALL    HOME  ;LINE FEED
0646 CD260A              CALL    RETN  ;CARR RETURN
0649 CD9D0A              CALL    PRIMM
```

```
064C 4D554C54         DB   'MULT. CONSTANT'
0650 2E20434F
0654 4E535441
0658 4E54
065A 00               DB      00H
065B CDBB06           CALL    PRMLC ;PRINT MULT CONSTANT
               ;  CHECK TEST ANTENNA COMPARATOR VALUE
065E 3A1221    CKTAN: LDA     TANCP ;LOAD VALUE
0661 FE14             CPI     20    ;COMPARE WITH DEFAULT
0663 CA7F06           JZ      CKPCT ;DEFAULT VALUE
               ;  TANCP HAS BEEN CHANGED
0666 CD000A           CALL    HOME  ;LINE FEED
0669 CD260A           CALL    RETN  ;CARR. RETURN
066C CD9D0A           CALL    PRIMM
066F 54455354         DB   'TEST COMPARE'
0673 20434F4D
0677 50415245
067B 00               DB   00H
067C CDE506           CALL    PRTAN ;PRINT VALUE
               ;  CHECK PERCENTAGE FOR ACTIVITY
067F 3A1D21    CKPCT: LDA     ACPCT ;LOAD VALUE
0682 FE50             CPI     80    ;COMPARE WITH DEFAULT
0684 CA3901           JZ      KEYB  ;DEFAULT VALUE
               ;  PERCENTAGE HAS BEEN CHANGED
0687 CD000A           CALL    HOME  ;LINE FEED
068A CD260A           CALL    RETN  ;CARR. RETURN
068D CD9D0A           CALL    PRIMM
0690 4143542E         DB   'ACT. PERCENT'
0694 20504552
0698 43454E54
069C 00               DB      00H
069D CDCE06           CALL    PRPCT ;PRINT VALUE
               ;  ALL PARAMETERS HAVE BEEN CHICKED
06A0 C33901           JMP     KEYB
               ;
               ;SUBROUTINE TO DISPLAY CEND
06A3 2A1F20    DCEN:  LHLD    CEND  ;LOAD HL REGISTERS
06A6 CD4C04           CALL    PHL   ;PRINT CEND
06A9 C33901           JMP     KEYB  ;RETURN
               ;
               ;SUBROUTINE TO DISPLAY CSTRT
06AC 2A1D20    DCST:  LHLD    CSTRT ;LOAD HL REGISTERS
06AF CD4C04           CALL    PHL   ;PRINT CSTRT
06B2 C33901           JMP     KEYB  ;RETURN
               ;
               ;SUBROUTINE TO DISPLAY THE MUITIPLIER CONSTANT
06B5 CDBB06    DMLC:  CALL    PRMLC
06B8 C33901           JMP     KEYB
06BB CD9D0A    PRMLC: CALL    PRIMM
06BE 3D20             DB   '= '
06C0 00               DB      00H
06C1 3A3F22           LDA     CMULT
06C4 CD2108           CALL    PRBYT
06C7 C9               RET
```

```
             ;
             ;SUBROUTINE TO DISPLAY ACTIVITY PERCENTAGE
06C8 CDCE06  DPCT:  CALL    PRPCT
06CB C33901         JMP     KEYB
06CE CD9D0A  PRPCT: CALL    PRIMM
06D1 3D20           DB    '= '
06D3 00             DB      00H
06D4 3A1D21         LDA     ACPCT
06D7 47             MOV  B, A     ;STORE VALUE IN B
06D8 3E64           MVI  A, 100
06DA 90             SUB  B        ;100- VALUE
06DB CD2108         CALL    PRBYT
06DE C9             RET
             ;
             ;SUBROUTINE TO DISPLAY TEST ANTENNA
             ;  COMPARATOR VALUE
06DF CDE506  DTAN:  CALL    PRTAN
06E2 C33901         JMP     KEYB
06E5 CD9D0A  PRTAN: CALL    PRIMM
06E8 3D20           DB    '= '
06EA 00             DB      00H
06EB 3A1221         LDA     TANCP
06EE CD2108         CALL    PRBYT
06F1 C9             RET
             ;
             ;SUBROUTINE TO DISPLAY THE CASSETTE
             ;  STORE INTERVAL
06F2 CDF806  DCSI:  CALL    PRCSI
06F5 C33901         JMP     KEYB
06F8 CD9D0A  PRCSI: CALL    PRIMM
06FB 3D20           DB    '= '
06FD 00             DB      00H
06FE 3A3622         LDA     CASI1
0701 CD2108         CALL    PRBYT
0704 C9             RET
             ;
             ;SUBROUTINE TO DISPLAY TPLEN
0705 CD0B07  DCTP:  CALL    PRCTP
0708 C33901         JMP     KEYB
070B CD9D0A  PRCTP: CALL    PRIMM ;PRINT EQUAL SIGN
070E 3D20           DB      '= '
0710 00             DB      00H
0711 3A2F20         LDA     TPLEN
0714 CD2108         CALL    PRBYT
0717 C9             RET
             ;
             ;SUBROUTINE TO DISPLAY DISTC
0718 3A1C21  DDIS:  LDA     DISTC ;LOAD VALUE
071B FE00           CPI     0     ;COMPARE TO DEFAULT
071D CA2607         JZ      DDDF  ;DEFAULT
0720 CD3107         CALL    PRDIS ;PRINT VALUE
0723 C33901         JMP     KEYB
0726 CD9D0A  DDDF:  CALL    PRIMM
0729 204F4646       DB    ' OFF'
```

```
072D 00                DB       00H
072E C33901            JMP      KEYB
0731 CD9D0A   PRDIS:   CALL     PRIMM
0734 3D20              DB    '= '
0736 00                DB       00H
0737 3A1C21            LDA      DISTC
073A CD2108            CALL     PRBYT
073D C9                RET
              ;
              ;SUBROUTINE TO DISPLAY ACTIVE CHANNELS
073E CD4407   DSAC:    CALL     DSAC1
0741 C33901            JMP      KEYB
0744 3A3620   DSAC1:   LDA      CHN1   ;HAVE CHANNELS BEEN SET?
0747 A7                ANA  A          ;SET FLAGS
0748 C26A07            JNZ      DSAC2  ;CHANNELS HAVE BEEN SET
              ;  THERE HAVE BEEN NO CHANNELS SET
074B CD000A            CALL     HOME   ;LINE FEED
074E CD260A            CALL     RETN
0751 CD9D0A            CALL     PRIMM
0754 4E4F2041          DB       'NO ACTIVE CHANNELS'
0758 43544956
075C 45204348
0760 414E4E45
0764 4C53
0766 00                DB       00H
0767 C33901            JMP      KEYB   ;RETURN
              ;  THERE HAVE BEEN CHANNELS SET
076A CD000A   DSAC2:   CALL     HOME   ;LINE FEED
076D CD260A            CALL     RETN   ;CARRIAGE RETURN
0770 CD9D0A            CALL     PRIMM
0773 41435449          DB       'ACTIVE CHANNELS'
0777 56452043
077B 48414E4E
077F 454C53
0782 00                DB       00H
0783 CD000A            CALL     HOME   ;LINE FEED
0786 CD260A            CALL     RETN   ;CARRIAGE RETURN
0789 0605              MVI  B,  05     ;LOAD COUNTER
078B 213620            LXI  H,  CHN1   ;POINT HL AT TABLE
078E 11C10C   CNEXT:   LXI  D,  CHTAB
0791 0E01              MVI  C,  01
0793 1A       CKTAB:   LDAX D          ;LOAD FIRST TABLE ENTRY
0794 BE                CMP  M          ;COMPARE WITH CHANNEL CODE
0795 CA9D07            JZ       FDCHN  ;FOUND ACTIVE CHANNEL
0798 0C                INR  C          ;INCREMENT CHANNEL NUMBER
0799 13                INX  D          ;CHECK NEXT TABLE ENTRY
079A C39307            JMP      CKTAB
              ;  CHANNEL NUMBER HAS BEEN FOUND
079D 79       FDCHN:   MOV  A, C       ;LOAD CHANNEL NUMBER
079E CD2108            CALL     PRBYT  ;PRINT CHANNEL NUMBER
07A1 CDE809            CALL     MOVR   ;LEAVE SPACE
07A4 23                INX  H          ;CHECK NEXT CHANNEL NUMBER
07A5 05                DCR  B          ;DECREMENT COUNTER
07A6 C8                RZ
```

```
07A7 7E              MOV  A, M     ;LOAD NEXT CHANNEL CODE
07A8 A7              ANA  A        ;SET FLAGS
07A9 C8              RZ
07AA C38E07          JMP     CNEXT ;DISPLAY NEXT CHANNEL
                ;
                ;SUBROUTINE TO DISPLAY CHANNEL
07AD CD9D0A    DCHN: CALL    PRIMM
07B0 3D20            DB      '= '
07B2 00              DB      00H
07B3 3A3520          LDA     CHNL
07B6 A7              ANA  A         ;SET FLAGS
07B7 CAC007          JZ      NONE   ;NO CHANNEL SET
                ;  A CHANNEL HAS BEEN SET
                ;  PRINT THE CHANNEL NUMBER
07BA CD2108          CALL    PRBYT ;PRINT CHANNEL NUMBER
07BD C33901          JMP     KEYB  ;RETURN
                ;  NO CHANNEL HAS BEEN SET
07C0 CD9D0A    NONE: CALL    PRIMM
07C3 4E4F4E45        DB      'NONE'
07C7 00              DB      00H
07C8 C33901          JMP     KEYB  ;RETURN
                ;
                ;SUBROUTINE TO DISPLAY PRINT INTERVAL
07CB 3A2F22    DPRI: LDA     PROF  ;READ PRINTER FLAG
07CE A7              ANA  A        ;SET FLAGS
07CF CAE107          JZ      DPRI1 ;PRINTER DISABLED
                ;  PRINTER HAS BEEN ENABLED,
                ;  PRINT THE PRINT INTERVAL
07D2 CD9D0A          CALL    PRIMM
07D5 3D20            DB   '= '
07D7 00              DB   00H
07D8 3A3122          LDA     PINT
07DB CD2108          CALL    PRBYT ;PRINT PRINT INTERVAL
07DE C33901          JMP     KEYB
                ;  PRINTER IS DISABLED
07E1 CD9D0A    DPRI1: CALL   PRIMM
07E4 204F4646        DB    ' OFF'
07E8 00              DB      00H
07E9 C33901          JMP     KEYB
                ;
                ;SUBROUTINE TO DISPLAY SCAN INTERVAL
07EC CD9D0A    DSCI: CALL    PRIMM
07EF 3D20            DB   '= '
07F1 00              DB   00H
07F2 3A3322          LDA     SCAN1
07F5 CD2108          CALL    PRBYT ;PRINT SCAN INTERVAL
07F8 C33901          JMP     KEYB
                ;
                ;SUBROUTINE TO DISPLAY PRINT ACTIVITY FLAG
07FB 3A1A21    DPOS: LDA     ACLOC
07FE A7              ANA  A         ;SET FLAGS
07FF CA0D08          JZ      DPOS2 ;PRINT POSITIONS
                ;  ACTIVITY TO BE PRINTED
0802 CD9D0A    DPOS1: CALL   PRIMM
```

```
0805 204F4646        DB   ' OFF'
0809 00              DB      00H
080A C33901          JMP     KEYB  ;RETURN
              ;  POSITIONS TO BE PRINTED
080D CD9D0A   DPOS2: CALL    PRIMM
0810 204F4E          DB   ' ON'
0813 00              DB      00H
0814 C33901          JMP     KEYB  ;RETURN
0817 3A1A21   DFLG:  LDA     ACLOC
081A A7              ANA  A          ;SET FLAGS
081B CA0208          JZ      DPOS1
081E C30D08          JMP     DPOS2

              ;
              ;SUBROUTINE TO PRINT ONE BYTE IN DECIMAL
              ;NUMBER IS ENTERED IN THE ACCUMULATOR
              ;THE BINARY NUMBER MUST BE CONVERTED
              ;TO DECIMAL, AND THEN TO ASCII AND
              ;THEN PRINTED.
0821 E5       PRBYT: PUSH H
0822 C5              PUSH B
0823 CD7608          CALL    HXDEC ;CONVERT TO DECIMAL
0826 4F              MOV  C, A     ;STORE IN REG C
              ;  PRINT MS DIGIT
0827 E6F0            ANI     0F0H  ;MASK OFF LS DIGIT
0829 0F              RRC
082A 0F              RRC
082B 0F              RRC
082C 0F              RRC
082D CDBC08          CALL    CONAS ;CONVERT TO ASCII
0830 2A1220          LHLD    POINT
0833 77              MOV  M, A     ;PUT MS DIGIT IN STACK
0834 CD320A          CALL    PRINT ;PRINT MS DIGIT
              ;  GET SECOND DIGIT
0837 23              INX  H
0838 79              MOV  A, C     ;LOAD CHANNEL NUMBER
              ;  PRINT SECOND DIGIT
0839 E60F            ANI     0FH   ;MASK OFF MS DIGIT
083B CDBC08          CALL    CONAS ;CONVERT TO ASCII
083E 77              MOV  M, A
083F CD320A          CALL    PRINT ;PRINT LS DIGIT
0842 C1              POP  B
0843 E1              POP  H
0844 C9              RET
```

```
                ;********************************************
                ;*                                          *
                ;*            UTILITY SUBROUTINES           *
                ;*                                          *
                ;********************************************
                ;
                ;SUBROUTINE TO MULTIPLY TWO 8 BIT INTEGERS
                ;  NUMBERS TO BE MULTIPLIED ARE LOADED INTO
                ;  THE B & C REGISTERS.  THE ANSWER IS
                ;  RETURNED IN THE BC REGISTER.
0845 D5         MULT8: PUSH D
0846 E5                PUSH H
0847 58                MOV  E, B
0848 0608              MVI  B, 08D   ;LOAD COUNTER WITH 8
084A AF                XRA  A        ;CLEAR ACCUMULATOR
084B 57                MOV  D, A
084C 67                MOV  H, A
084D 6F                MOV  L, A     ;CLEAR REGISTERS
084E 79         MULT:  MOV  A, C     ;LOAD MULTIPLIER
084F 0F                RRC           ;SHIFT D0 INTO CARRY
0850 4F                MOV  C, A     ;STORE SHIFTED MULTIPLIER
0851 D25508            JNC      NOMUL ;BIT IS ZERO
0854 19                DAD  D        ;ADD MULTIPLICAND TO HL
0855 7B         NOMUL: MOV  A, E     ;LOAD LS MULTIPLICAND
0856 17                RAL           ;SHIFT LEFT, D7 TO CY
0857 5F                MOV  E, A
0858 7A                MOV  A, D     ;LOAD MS MULTIPLICAND
0859 17                RAL           ;CY TO D0
085A 57                MOV  D, A
085B 05                DCR  B        ;DECREMENT COUNTER
085C C24E08            JNZ      MULT ;NOT FINISHED YET
085F 4D                MOV  C, L
0860 44                MOV  B, H
0861 E1                POP  H
0862 D1                POP  D
0863 C9                RET
                ;
                ;SUBROUTINE TO CONVERT DECIMAL NUMBER IN
                ;ACCUMULATOR TO HEX NUMBER
                ;THE RESULT IS RETURNED IN THE ACCUMULATOR
0864 C5         DECHX: PUSH B
0865 47                MOV  B, A     ;TEMP STORAGE
0866 E60F              ANI      0FH  ;MASK HIGH DIGIT
0868 4F                MOV  C, A     ;STORE LS DIGIT
0869 78                MOV  A, B
086A E6F0              ANI      0F0H ;MASK LOW DIGIT
086C 0F                RRC
086D 0F                RRC           ;MULTIPLIED BY 4
086E 47                MOV  B, A     ;STORE IN B
086F 0F                RRC
0870 0F                RRC
0871 80                ADD  B        ;ADD ORIGINAL NUMBER
0872 07                RLC           ;MULTIPLY BY 2
0873 81                ADD  C        ;ADD LS DIGIT
```

```
0874 C1               POP  B
0875 C9               RET
               ;
               ;SUBROUTINE TO CONVERT HEX NUMBER IN ACC.
               ;TO A DECIMAL NUMBER
               ;THE RESULT IS RETURNED IN THE ACCUMULATOR
               ;  CARRY BIT IS SET IF THERE HAS BEEN AN ERROR
0876 C5        HXDEC: PUSH B
0877 D5               PUSH D
0878 1E00             MVI  E, 00H
087A 0E00             MVI  C, 00H
087C 0608             MVI  B, 08D    ;LOAD COUNTER
087E 07        ROT:   RLC            ;ROTATE D7 INTO CARRY
087F 57               MOV  D, A      ;STORE ACCUMULATOR
0880 79               MOV  A, C      ;LOAD ANSWER
0881 8B               ADC  E         ;ADD CARRY
0882 27               DAA            ;DECIMAL ADJUST
0883 DA9908           JC      EREX   ;ERROR EXIT
0886 05               DCR  B         ;DECREMENT COUNTER
0887 CA9408           JZ      END1   ;DONE?
088A 87               ADD  A         ;MULTIPLY BY 2
088B 27               DAA            ;DECIMAL ADJUST
088C DA9908           JC      EREX   ;ERROR EXIT
088F 4F               MOV  C, A      ;STORE ANSWER
0890 7A               MOV  A, D      ;LOAD ACCUMULATOR
0891 C37E08           JMP     ROT    ;NOT DONE YET
0894 D1        END1:  POP  D
0895 C1               POP  B
0896 37               STC
0897 3F               CMC            ;CLEAR CARRY
0898 C9               RET
0899 D1        EREX:  POP  D
089A C1               POP  B
089B 37               STC            ;SET CARRY
089C C9               RET
               ;
               ;SUBROUTINE TO CONVERT THE ASCII CHAR. IN THE
               ;ACCUM. TO ITS CORRESPONDING HEX NUMBER AND
               ;RETURN THE ANSWER IN THE ACCUMULATOR.
               ;THE CARRY BIT IS SET IF THERE
               ;HAS BEEN AN ERROR.
089D C5        CONHX: PUSH    B      ;STORE BC REGISTERS
089E E5               PUSH    H      ;STORE HL REGISTERS
089F 4F               MOV  C, A      ;STORE CHARACTER IN C
08A0 0610             MVI  B, 10H    ;LOAD COUNTER
08A2 210109           LXI  H, ASCHX  ;LOAD TABLE ADDRESS
08A5 7E        TAGIN: MOV  A, M      ;LOAD TABLE DATA
08A6 B9               CMP     C      ;COMPARE WITH CHARACTER
08A7 CAB708           JZ      FDHEX  ;CORRECT  CHARACTER
08AA 05               DCR     B      ;DECREMENT COUNTER
08AB CAB308           JZ      ERLV   ;ERROR EXIT
08AE 23               INX     H
08AF 23               INX     H      ;POINT AT NEXT TABLE ENTRY
08B0 C3A508           JMP     TAGIN  ;CHECK NEXT TABLE ENTRY
```

```
08B3 37        ERLV:  STC               ;SET CARRY
08B4 C3B908           JMP      LEVE     ;EXIT FROM SUBROUTINE
08B7 23        FDHEX: INX      H        ;POINT AT HEX NUMBERS
08B8 7E               MOV   A, M        ;LOAD HEX NUMBER INTO A
08B9 E1        LEVE:  POP      H        ;RESTORE HL REGISTER
08BA C1               POP      B        ;RESTORE BC REGISTERS
08BB C9               RET
               ;
               ;SUBROUTINE TO CONVERT THE HEX NUMBER IN THE
               ;ACC. TO ITS CORRESPONDING ASCII CHARACTER AND
               ;RETURN THE ANSWER IN THE ACCUMULATOR.
               ;CARRY BIT IS SET IF THERE HAS BEEN AN ERROR.
08BC C5        CONAS: PUSH     B        ;STORE BC REGISTERS
08BD E5               PUSH     H        ;STORE HL REGISTERS
08BE 4F               MOV   C, A        ;STORE NUMBER IN C
08BF 0610             MVI   B, 10H      ;LOAD COUNTER
08C1 210209           LXI   H, ASCHX+1 ;LOAD TABLE ADDRESS
08C4 7E        TAGN1: MOV   A, M        ;LOAD TABLE DATA
08C5 B9               CMP      C        ;COMPARE WITH NUMBER
08C6 CAD608           JZ       FDASC ;CORRECT  NUMBER
08C9 05               DCR      B        ;DECREMENT COUNTER
08CA CAD208           JZ       ERLV1 ;ERROR EXIT
08CD 23               INX      H
08CE 23               INX      H        ;POINT AT NEXT TABLE ENTRY
08CF C3C408           JMP      TAGN1 ;CHECK NEXT TABLE ENTRY
08D2 37        ERLV1: STC               ;SET CARRY
08D3 C3D808           JMP      LEAV1 ;EXIT FROM SUBROUTINE
08D6 2B        FDASC: DCX      H        ;POINT AT ASCII CHARACTER
08D7 7E               MOV   A, M        ;LOAD ASCII CHAR. INTO A
08D8 E1        LEAV1: POP      H        ;RESTORE HL REGISTER
08D9 C1               POP      B        ;RESTORE BC REGISTERS
08DA C9               RET
               ;
               ;SUBROUTINE TO RECEIVE TWO DECIMAL DIGITS
               ;FROM THE KEYBOARD
               ;  AND CONVERT THE NUMBER TO HEX
               ;  THE RESULT IS RETURNED IN THE ACCUMULATOR
08DB C5        DECKY: PUSH B
08DC CDC701           CALL     CHKEY ;READ KEYBOARD
08DF FE20             CPI      ' '      ;IS CHARACTER A SPACE
08E1 C29E09           JNZ      ERROR ;NOT A SPACE
08E4 CDC701           CALL     CHKEY ;READ MS DIGIT
08E7 CD9D08           CALL     CONHX ;CONVERT TO DECIMAL
08EA DA9E09           JC       ERROR ;ERROR EXIT
08ED 07               RLC
08EE 07               RLC
08EF 07               RLC
08F0 07               RLC               ;ROTATE INTO HIGH NIBBLE
08F1 47               MOV   B, A        ;STORE IN REG B
08F2 CDC701           CALL     CHKEY ;RECEIVE LS DIGIT
08F5 CD9D08           CALL     CONHX ;CONVERT TO DECIMAL
08F8 DA9E09           JC       ERROR ;ERROR EXIT
08FB B0               ORA   B           ;ADD LS DIGIT
08FC CD6408           CALL     DECHX ;CONVERT TO HEX
```

```
08FF C1             POP  B
0900 C9             RET
             ;
             ;LOOK-UP TABLE FOR ASCII-HEX CONV. ROUTINES.
0901 30003101 ASCHX: DB  30H,00H,31H,01H
0905 32023303       DB   32H,02H,33H,03H
0909 34043505       DB   34H,04H,35H,05H
090D 36063707       DB   36H,06H,37H,07H
0911 38083909       DB   38H,08H,39H,09H
0915 410A420B       DB   41H,0AH,42H,0BH
0919 430C440D       DB   43H,0CH,44H,0DH
091D 450E460F       DB   45H,0EH,46H,0FH
             ;
             ;SUBROUTINE TO LOAD THE ADD. IN THE HL REGISTER
             ;INTO THE FIRST FOUR PRINT POSITIONS.
0921 7C      HLSTK: MOV  A, H     ;GET H REGISTER
0922 E6F0           ANI     0F0H  ;MASK OFF LOW NIBBLE
0924 0F             RRC
0925 0F             RRC
0926 0F             RRC
0927 0F             RRC           ;ROTATE INTO LOW NIBBLE
0928 CDBC08         CALL    CCNAS ;CONVERT TO ASCII
092B 320020         STA     CHR1  ;LOAD INTO CHR1
092E 7C             MOV  A, H     ;GET H REGISTER
092F E60F           ANI     0FH   ;MASK OFF HIGH NIBBLE
0931 CDBC08         CALL    CCNAS ;CONVERT TO ASCII
0934 320120         STA     CHR2  ;LOAD INTO CHR2
0937 7D             MOV  A, L     ;GET L REGISTER
0938 E6F0           ANI     0F0H  ;MASK OFF LOW NIBBLE
093A 0F             RRC
093B 0F             RRC
093C 0F             RRC
093D 0F             RRC           ;ROTATE INTO LOW NIBBLE
093E CDBC08         CALL    CCNAS ;CONVERT TO ASCII
0941 320220         STA     CHR3  ;LOAD INTO CHR3
0944 7D             MOV  A, L     ;GET L REGISTER
0945 E60F           ANI     0FH   ;MASK OFF HIGH NIBBLE
0947 CDBC08         CALL    CCNAS ;CONVERT TO ASCII
094A 320320         STA     CHR4  ;LOAD INTO CHR4
094D C9             RET
             ;
             ;SUBROUTINE TO RECEIVE FOUR ASCII CHARACTERS,
             ;CONVERT THE NUMBER TO HEX, AND STORE IT
             ;IN THE HL REGISTER.
             ;BECAUSE OF THE ERROR EXIT, THIS SUBROUTINE
             ;CAN ONLY BE CALLED BY THE MAIN PROGRAM.
094E CDC701  GTADD: CALL    CHKEY ;READ KEYBOARD
0951 CD9D08         CALL    CCNHX ;CONVERT CHARACTER TO HEX
0954 DA8109         JC      ERX   ;ERROR EXIT
0957 07             RLC
0958 07             RLC
0959 07             RLC
095A 07             RLC           ;MOVE TO HIGH ORDER NIBBLE
095B 67             MOV  H, A     ;PUT IN H REGISTER
```

```
095C CDC701          CALL    CHKEY ;READ KEYBCARD
095F CD9D08          CALL    CONHX ;CCNVERT TO HEX
0962 DA8109          JC      ERX   ;ERROR EXIT
0965 B4              ORA     H
0966 67              MOV  H, A     ;MCVE TO LCW ORDER NIBBLE
0967 CDC701          CALL    CHKEY ;READ KEYBOARD
096A CD9D08          CALL    CONHX ;CCNVERT CHARACTER TO HEX
096D DA8109          JC      ERX   ;ERROR EXIT
0970 07              RLC
0971 07              RLC
0972 07              RLC
0973 07              RLC           ;MOVE TO HIGH ORDER NIBBLE
0974 6F              MOV  L, A     ;PUT IN L REGISTER
0975 CDC701          CALL    CHKEY ;READ KEYBOARD
0978 CD9D08          CALL    CONHX ;CCNVERT TO HEX
097B DA8109          JC      ERX   ;ERROR EXIT
097E B5              ORA     L
097F 6F              MOV  L, A     ;MOVE TO LCW ORDER NIBBLE
0980 C9              RET
0981 33        ERX:  INX     SP
0982 33              INX     SP    ;RESTORE THE STACK
0983 C39E09          JMP     ERROR ;ERROR EXIT
               ;
               ;SUBROUTINE TO ADD 21 TO THE HL REGISTER
0986 D5        AD21:  PUSH D
0987 111500           LXI  D, 21
098A 19               DAD  D         ;HL + DE
098B D1               POP  D
098C C9               RET
               ;
               ;SUBROUTINE TO INCREMENT DE 21 TIMES
098D F5        INDE:  PUSH PSW
098E E5               PUSH H
098F 211500           LXI  H, 21
0992 19               DAD  D
0993 EB               XCHG
0994 E1               POP  H
0995 F1               PCP  PSW
0996 C9               RET
               ;
               ;SUBROUTINE TO ADD 882 TO THE HL REGISTER
0997 D5        AD882: PUSH D
0998 117203           LXI  D, 882
099B 19               DAD  D
099C D1               POP  D
099D C9               RET
               ;
               ;SUBROUTINE TO PRINT ERROR MESSAGE
099E CD000A    ERROR: CALL    HOME  ;LINE FEED
09A1 CD260A           CALL    RETN  ;CARRIAGE RETURN
09A4 CD9D0A           CALL    PRIMM ;PRINT MESSAGE
09A7 4552524F         DB      'ERROR'
09AB 52
09AC 00               DB      00H
```

```
09AD C33901          JMP     KEYB  ;RETURN
               ;*********************************************
               ;*                                           *
               ;*      STEPPER MOTOR CONTROL PROGRAMS       *
               ;*                                           *
               ;*********************************************
               ;PRPOS- ADDRESS IN THE STEPPER MOTOR
               ;       TABLE WHICH CORRESPONDS TO THE
               ;       PRESENT PRINTHEAD POSITION.
               ;POINT- ADDRESS IN THE PRINT STACK
               ;       WHICH CORRESPONDS TO THE
               ;       PRESENT PRINTHEAD POSITION.
               ;
               ;STEPPER MOTOR DRIVE SIGNALS
09B0 060A0905         DB   06H,0AH,09H,05H
09B4 060A0905  STEP:  DB   06H,0AH,09H,05H
09B8 060A0905         DB   06H,0AH,09H,05H
09BC 060A0905         DB   06H,0AH,09H,05H
09C0 060A0905         DB   06H,0AH,09H,05H
09C4 060A0905         DB   06H,0AH,09H,05H
09C8 060A0905         DB   06H,0AH,09H,05H
09CC 060A0905         DB   06H,0AH,09H,05H
               ;
               ;SUBROUTINE TO MOVE THE PRINTHEAD
               ;ONE POSITION TO THE LEFT
09D0 E5        MOVL:  PUSH    H     ;STORE HL REGISTERS
               ;  POINT AT NEXT CHARACTER IN PRINT STACK
09D1 2A1220           LHLD    POINT ;CONTENTS OF POINT LOADED
09D4 2B               DCX     H
09D5 221220           SHLD    POINT ;DECREMENT POINT
               ;  POINT AT NEXT STEPPER MOTOR SIGNALS
09D8 2A1420           LHLD    PRPOS ;LOAD PRINT POSITION
09DB 2B               DCX     H     ;MOVE ONE LOCATION LEFT
09DC 221420           SHLD    PRPOS
               ;  OUTPUT STEPPER MOTOR CONTROL SIGNALS
               ;  AND DELAY 20 MSEC.
09DF CDA00B           CALL    DEL20 ;DELAY 20 MSEC.
09E2 3E00             MVI  A, 00H
09E4 D304             OUT     STPER ;TURN OFF STEPPER CONTROLS
09E6 E1               POP     H     ;RESTORE HL REGISTERS
09E7 C9               RET
               ;
               ;SUBROUTINE TO MOVE THE PRINTHEAD
               ;ONE POSITION TO THE RIGHT
09E8 E5        MOVR:  PUSH    H     ;STORE HL REGISTERS
               ;  POINT AT NEXT CHARACTER IN PRINT STACK
09E9 2A1220           LHLD    POINT ;CONTENTS OF POINT LOADED
09EC 23               INX     H
09ED 221220           SHLD    POINT ;INCREMENT POINT
               ;  POINT AT NEXT STEPPER MOTOR SIGNALS.
09F0 2A1420           LHLD    PRPOS ;LOAD PRINT POSITION
09F3 23               INX     H     ;MOVE ONE LOCATION RIGHT
09F4 221420           SHLD    PRPOS
               ;  OUTPUT STEPPER MOTOR CONTROL SIGNALS
```

```
                    ;  AND DELAY 20 MSEC.
09F7 CDA00B                CALL      DEL20 ;DELAY 20 MSEC.
09FA 3E00                  MVI   A,  00H
09FC D304                  OUT       STPER ;TURN OFF STEPPER CONTROLS
09FE E1                    POP       H     ;RESTORE HL REGISTERS
09FF C9                    RET
                    ;
                    ;SUBROUTINE TO MOVE PRINTHEAD TO HOME POSITION
                    ;NOTE: IF THE PRINTHEAD IS ALREADY IN THE HOME
                    ;      POSTION, THIS SUBROUTINE WILL CAUSE A
                    ;      LINE FEED.
                    ;THE HOME POITION IS 2 POITIONS TO THE LEFT
                    ;OF THE END OF THE CARRIAGE.
                    ;THE EOT SWITCH CLOSES WHEN THE PRINTHEAD
                    ;REACHES THE END OF THE CARRIAGE.
0A00 E5             HOME:  PUSH      H      ;STORE HL REGISTERS
0A01 CDE809         HOME1: CALL      MOVR   ;MOVE PRINTHEAD RIGHT
  1               +        READ      EOT    ;CHECK EOT SWITCH
0A04 1 D5         +        PUSH      D      ;SAVE DE REGISTERS
0A05 1 1601       +        MVI   D,  EOT   ;STORE SWITCH NUMBER IN D
0A07 1 DB06       +        IN        SWCH   ;READ THE SWITCH INPUTS
0A09 1 1F         +NO:     RAR              ;ROTATE D0 INTO CARRY
0A0A 1 15         +        DCR   D          ;DECREMENT COUNTER
0A0B 1 C2090A +            JNZ       NO     ;NOT CORRECT SWITCH YET
0A0E 1 D1         +        POP       D      ;RESTORE DE REGISTERS
0A0F DA010A                JC        HOME1  ;EOT NOT CLOSED YET
0A12 CDD009                CALL      MOVL   ;PRINTHEAD TO POSITION A
0A15 CDD009                CALL      MOVL   ;PRINTHEAD TO HOME
                    ;  RESET POINTERS
0A18 211220                LXI   H,  CHR1+18D ;INITIALIZE POINT
0A1B 221220                SHLD      POINT ;INITIALIZE POINT
0A1E 21C609                LXI   H,  STEP+18D
0A21 221420                SHLD      PRPOS ;INITIALIZE PRPOS
0A24 E1                    POP       H      ;RESTORE HL REGISTERS
0A25 C9                    RET
                    ;
                    ;SUBROUTINE TO MOVE PRINTHEAD FROM HOME
                    ;POSITION TO FIRST PRINT POSITION.
                    ;  FIRST PRINT POITION IS 18 POSITIONS TO
                    ;  THE LEFT OF THE HOME POSITION.
0A26 D5             RETN:  PUSH      D      ;STORE DE REGISTERS
0A27 1612                  MVI   D,  18D    ;LOAD COUNTER WITH 18
0A29 CDD009         MORE:  CALL      MOVL   ;MOVE PRINTHEAD LEFT
0A2C 15                    DCR       D      ;DECREMENT COUNTER
0A2D C2290A                JNZ       MORE   ;NOT AT POSITION 1 YET
0A30 D1                    POP       D      ;RESTORE DE REGISTERS
0A31 C9                    RET
```

```
                 ;*******************************************
                 ;*                                         *
                 ;*          PRINTER CONTROL PROGRAMS       *
                 ;*                                         *
                 ;*******************************************
                 ;
                 ;SUBROUTINE TO PRINT THE CHARACTER WHICH IS
                 ;POINTED TO BY POINT. THE PRINTHEAD IS THEN
                 ;SHIFTED ONE POSTION TO THE RIGHT.
0A32 E5          PRINT: PUSH    H     ;STORE HL REGISTERS
0A33 2A1220             LHLD    POINT ;POINT HL AT DATA
0A36 7E                 MOV  A, M     ;LOAD ASCII DATA INTO A
0A37 D305               OUT     ASCII ;OUTPUT TO CHAR GENERATOR
                 ;  COUNTER NUMBER 0 IS USED
                 ;  TO GENERATE THE PRINT STROBE
0A39 3E00               MVI  A, 00H
0A3B D304               OUT     STPER ;CLEAR FLIP FLOP
0A3D 3E40               MVI  A, 40H
0A3F D304               OUT     STPER ;ENABLE FLIP FLOP
0A41 3E32               MVI  A, 32H
0A43 D300               OUT     MODEA ;SET COUNTER MODE
0A45 3E2F               MVI  A, 2FH
0A47 D303               OUT     TIMA0 ;LOAD LOW ORDER COUNT
0A49 3E0D               MVI  A, 0DH
0A4B D303               OUT     TIMA0 ;LOAD HIGH ORDER COUNT
                 ;  START PRINT STROBE
0A4D 3EC0               MVI  A, 0C0H
0A4F D304               OUT     STPER ;TRIGGER STROBE
                 ;  CHECK TIMER
0A51 1          +CHKB:  READ    TEST  ;CHECK TIMER
0A51 1 D5       +       PUSH    D     ;SAVE DE REGISTERS
0A52 1 1608     +       MVI  D, TEST  ;STORE SWITCH NUMBER IN D
0A54 1 DB06     +       IN      SWCH  ;READ THE SWITCH INPUTS
0A56 1 1F       +NO:    RAR           ;ROTATE D0 INTO CARRY
0A57 1 15       +       DCR  D        ;DECREMENT COUNTER
0A58 1 C2560A +         JNZ     NO    ;NOT CORRECT SWITCH YET
0A5B 1 D1       +       POP     D     ;RESTORE DE REGISTERS
0A5C DA510A             JC      CHKB  ;TIMER NOT DONE
                 ;  TIMER IS FINISHED
                 ;  GENERATE 6 MSEC DELAY
                 ;  TO ALLOW THE PRINTHEAD TO COOL OFF
0A5F 3E00               MVI  A, 00H
0A61 D304               OUT     STPER ;CLEAR FLIP FLOP
0A63 3E40               MVI  A, 40H
0A65 D304               OUT     STPER ;ENABLE FLIP FLOP
0A67 3E78               MVI  A, 78H
0A69 D300               OUT     MODEA ;SET COUNTER MODE
0A6B 3EDC               MVI  A, 0DCH
0A6D D302               OUT     TIMA1 ;LOAD LOW ORDER COUNT
0A6F 3E05               MVI  A, 05H
0A71 D302               OUT     TIMA1 ;LOAD HIGH ORDER COUNT
                 ;  CHECK TIMER
0A73 1          +CHKC:  READ    TEST  ;CHECK THE TIMER
0A73 1 D5       +       PUSH    D     ;SAVE DE REGISTERS
```

```
0A74 1 1608 +        MVI  D, TEST  ;STORE SWITCH NUMBER IN D
0A76 1 DB06 +        IN      SWCH  ;READ THE SWITCH INPUTS
0A78 1 1F   +NO:     RAR           ;ROTATE D0 INTO CARRY
0A79 1 15   +        DCR  D        ;DECREMENT COUNTER
0A7A 1 C2780A +      JNZ     NO    ;NOT CORRECT SWITCH YET
0A7D 1 D1   +        POP     D     ;RESTORE DE REGISTERS
0A7E DA730A          JC      CHKC  ;TIMER NOT DONE
               ;  TIMER IS DONE, MOVE PRINTHEAD
0A81 CDE809          CALL    MOVR  ;MOVE PRINTHEAD RIGHT
0A84 E1              POP     H     ;RESTORE HL REGISTERS
0A85 C9              RET
               ;
               ;SUBROUTINE TO PRINT THE ENTIRE CONTENTS
               ;OF THE PRINT STACK.
               ;THE PRINTER IS INITIALLY IN THE HOME POSITION
0A86 C5        PSTAK: PUSH   B     ;STORE BC REGISTERS
0A87 E5               PUSH   H     ;STORE HL REGISTERS
0A88 CD260A           CALL   RETN  ;MOVE TO FIRST POSITION
0A8B 0612             MVI  B, 18D  ;LOAD COUNTER
0A8D CD320A    MOR:   CALL   PRINT ;PRINT ONE CHARACTER
0A90 05               DCR    B     ;DECREMENT COUNTER
0A91 C28D0A           JNZ    MOR   ;FULL LINE NOT PRINTED YET
               ;  PRINTING IS FINISHED, RETURN PRINTHEAD TO
               ;  THE HOME POSITION AND CLEAR THE PRINT STACK
0A94 CD000A           CALL   HOME  ;LINE FEED
0A97 CDB70A           CALL   CLPSK ;CLEAR THE PRINT STACK
0A9A E1               POP    H     ;RESTORE HL REGISTERS
0A9B C1               POP    B     ;RESTORE BC REGISTERS
0A9C C9               RET
               ;
               ;SUBROUTINE TO PRINT THE DATA IMM. FOLLOWING
               ;THE CALL INSTRUCTION.  THE IMM. DATA MUST
               ;BE FOLLOWED BY A ZERO BYTE.
0A9D E3        PRIMM: XTHL         ;EXCHANGE STACK WITH HL
0A9E EB               XCHG         ;EXCHANGE HL AND DE
0A9F E5               PUSH   H     ;STORE CONTENTS OF DE
0AA0 2A1220           LHLD   POINT ;POINT HL AT PRINT STACK
               ;  DE NOW POINTS TO THE ADDRESS AFTER THE CALL
               ;  HL POINTS AT THE PRINT STACK
0AA3 1A        PMORE: LDAX   D     ;READ IMMEDIATE DATA
0AA4 FE00             CPI    00H   ;COMPARE WITH ZERO
               ;  VALID ASCII DATA, THEREFORE PRINT IT
0AA6 CAB20A           JZ     EXP   ;PRINTING IS FINISHED
0AA9 77               MOV  M, A    ;STORE IN PRINT STACK
0AAA CD320A           CALL   PRINT ;PRINT THE CHARACTER
0AAD 23               INX    H
0AAE 13               INX    D     ;POINT AT NEW DATA
0AAF C3A30A           JMP    PMORE
               ;  A ZERO BYTE HAS BEEN FOUND IN THE DATA
               ;  TABLE. THIS SIGNIFIES THE END OF THE
               ;  TABLE. THE NEXT INSTRUCTION BEGINS AT
               ;  THE ADDRESS IMMEDIATELY AFTER THE
               ;  ZERO BYTE.
0AB2 13        EXP:   INX    D     ;POINT AT NEXT INSTRUCTION
```

```
0AB3 EB              XCHG            ;EXCHANGE HL AND DE
                ;  HL NOW CONTAINS THE PROPER RETURN ADDRESS
0AB4 D1              POP     D       ;RESTORE DE REGISTER
                ;  PUT RETURN ADDRESS ON THE STACK
0AB5 E3              XTHL            ;STORE NEW RETURN ADDRESS
0AB6 C9              RET
                ;
                ;SUBROUTINE TO CLEAR THE PRINT STACK
0AB7 C5         CLPSK: PUSH    B
0AB8 E5              PUSH    H
0AB9 0612            MVI  B, 18D   ;LOAD COUNTER WITH 18
0ABB 210020          LXI  H, CHR1  ;POINT HL AT PRINT STACK
0ABE 3600       NCLR: MVI  M, 00H   ;CLEAR LOCATION
0AC0 23              INX     H     ;INCREMENT POINTER
0AC1 05              DCR     B     ;DECREMENT COUNTER
0AC2 C2BE0A          JNZ     NCLR  ;NOT ALL CLEARED YET
0AC5 E1              POP     H
0AC6 C1              POP     B
0AC7 C9              RET
                ;
                ;SUBROUTINE TO ALLOW PRINTING ON THE PRINTER
                ;CR (CTRL M) WILL CAUSE CARR RET AND LINE FEED
                ;BS (CTRL H) WILL CAUSE A BACKSPACE
                ;ETX (CTRL C) RETURN CONTROL TO THE MONITOR
0AC8 CD000A     PRT:   CALL    HOME  ;LINE FEED
0ACB CD260A            CALL    RETN  ;CARRIAGE RETURN
                ;  A CHARACTER IS RECEIVED FROM THE KEYBOARD
0ACE CDC701     RECEV: CALL    CHKEY ;READ KEYBOARD
0AD1 FE0D              CPI     CRET
0AD3 CAE30A            JZ      CARET ;CARRIAGE RETURN
                ;  NOT A CARRIAGE RETURN
0AD6 FE08              CPI     BAKSP
0AD8 CAEC0A            JZ      BACKS ;BACKSPACE
                ;  NOT A BACKSPACE
0ADB FE03              CPI     ETX
0ADD CA3901            JZ      KEYB  ;RETURN FROM ROUTINE
0AE0 C3CE0A            JMP     RECEV ;RECEIVE ANOTHER CHARACTER
0AE3 CD000A     CARET: CALL    HOME  ;LINE FEED
0AE6 CD260A            CALL    RETN  ;CARRIAGE RETURN
0AE9 C3CE0A            JMP     RECEV ;RECEIVE ANOTHER CHARACTER
0AEC CDD009     BACKS: CALL    MOVL  ;BACKSPACE
0AEF CDD009            CALL    MOVL
0AF2 C3CE0A            JMP     RECEV ;RECEIVE NEXT CHARACTER
```

```
                    ;*******************************************
                    ;*                                         *
                    ;*        ANTENNA CONTROL PROGRAMS         *
                    ;*                                         *
                    ;*******************************************
                    ;
                    ;SUBROUTINE TO TURN ON A SELECTED ANTENNA RELAY
0AF5 CDC701         ANT:   CALL    CHKEY ;READ KEYBOARD
0AF8 FE20                  CPI     ' '   ;IS CHARACTER A SPACE
0AFA C29E09                JNZ     ERROR ;ERROR EXIT
                    ;  A SPACE HAS BEEN RECEIVED AFTER THE COMMAND
                    ;  NOW RECEIVE AN X OR A Y
0AFD CDC701                CALL    CHKEY ;READ KEYBOARD
0B00 FE58                  CPI     'X'   ;IS CHARACTER AN X
0B02 C20A0B                JNZ     CY    ;NOT AN X
0B05 0E00                  MVI  C, 00H   ;RESET FLAG
0B07 C3110B                JMP     GNUMB ;RECEIVE ANTENNA NUMBER
0B0A FE59           CY:    CPI     'Y'   ;IS CHARACTER A Y
0B0C C29E09                JNZ     ERROR ;ERROR EXIT
0B0F 0EFF                  MVI  C, 0FFH  ;SET FLAG
                    ;  C IS SET IF Y WAS RECEIVED
                    ;  C IS RESET IF X WAS RECEIVED
                    ;  NOW RECEIVE THE ANTENNA NUMBER
0B11 CDC701         GNUMB: CALL    CHKEY ;READ KEYBOARD
0B14 CD9D08                CALL    CONHX ;CONVERT TO DECIMAL
0B17 DA9E09                JC      ERROR ;ERROR EXIT
0B1A 07                    RLC
0B1B 07                    RLC
0B1C 07                    RLC
0B1D 07                    RLC           ;ROTATE INTO MS NIBBLE
0B1E 47                    MOV  B, A     ;STORE IN REG B
                    ;  FIRST DIGIT HAS BEEN RECEIVED.
0B1F CDC701                CALL    CHKEY ;READ KEYBOARD
0B22 CD9D08                CALL    CONHX ;CONVERT TO DECIMAL
0B25 DA9E09                JC      ERROR ;ERROR EXIT
                    ;  A CONTAINS THE LS NIBBLE
                    ;  B CONTAINS THE MS NIBBLE
0B28 B0                    ORA  B        ;FORM 2 DIGIT NUMBER
0B29 CD6408                CALL    DECHX ;CONVERT TO HEX
0B2C 47                    MOV  B, A     ;TEMPORARILY STORE IN B
                    ;  CHECK IF X OR Y ANTENNAS
0B2D 79                    MOV  A, C     ;CHECK FLAG
0B2E A7                    ANA  A        ;SET FLAGS
0B2F C23A0B                JNZ     OUTY
                    ;  X ANTENNAS ARE TO BE SELECTED
0B32 78                    MOV  A, B     ;PUT CHARACTER IN A
                    ;  SET THE X/Y CONTROL LINE
0B33 F680                  ORI     80H   ;ENABLE X ANTENNAS
0B35 D310                  OUT     ANTA  ;TURN ON RELAY
0B37 C33D0B                JMP     INADC
                    ;  Y ANTENNAS ARE TO BE USED
0B3A 78             OUTY:  MOV  A, B     ;PUT CHARACTER IN A
0B3B D310                  OUT     ANTA  ;TURN ON RELAY
                    ;  THE ANTENNA HAS BEEN TURNED ON
```

51

```
0B3D CD9D0A  INADC: CALL    PRIMM
0B40 3D             DB      '='
0B41 00             DB      00H
              ;  DELAY TIME FOR RADIO SETTLING (1 MSEC)
0B42 117D00         LXI  D, 125D  ;LOAD COUNTER
0B45 1B      DELMS: DCX  D
0B46 7B             MOV  A, E
0B47 A7             ANA  A
0B48 C2450B         JNZ     DELMS
              ;  DELAY IS OVER
              ;  START CONVERSION
0B4B D312           OUT     STCON ;START CONVERSION
              ;  WAIT FOR AN EOC SIGNAL
0B4D 1      +RDADC: READ    EOC   ;CONVERSION DONE?
0B4D 1 D5    +      PUSH    D     ;SAVE DE REGISTERS
0B4E 1 1604 +       MVI  D, EOC  ;STORE SWITCH NUMBER IN D
0B50 1 DB06 +       IN      SWCH  ;READ THE SWITCH INPUTS
0B52 1 1F   +NO:    RAR           ;ROTATE D0 INTO CARRY
0B53 1 15   +       DCR  D        ;DECREMENT COUNTER
0B54 1 C2520B +     JNZ     NO    ;NOT CORRECT SWITCH YET
0B57 1 D1   +       POP     D     ;RESTORE DE REGISTERS
0B58 D24D0B         JNC     RDADC ;CHECK AGAIN
              ;  CONVERSION IS FINISHED
0B5B DB13           IN      ADCIN ;READ A/D CONVERTER
0B5D A7             ANA  A        ;SET FLAGS
0B5E F2630B         JP      NNEG
              ;  SIGNAL IS NEGATIVE, SET OUTPUT TO 0
              ;  NOTE: THE INPUT FROM THE A/D
              ;        IS COMPLEMENTARY.
0B61 3E7F           MVI  A, 7FH
0B63 2F      NNEG:  CMA           ;OUTPUT IS COMPLEMENTARY
0B64 D680           SUI     80H
              ;  INPUT IS NOW A HEX NUMBER FROM 00 TO 7F
0B66 47             MOV  B, A     ;STORE IN REG B
              ;  MULTIPLY BY CORRECT FACTOR TO OBTAIN
              ;  THE VALUE OF THE INPUT IN VOLTS
0B67 0E65           MVI  C, 65H   ;LOAD MULT. FACTOR
0B69 CD4508         CALL    MULT8 ;MULTIPLY
0B6C 78             MOV  A, B     ;LOAD MS BYTE
0B6D CD7608         CALL    HXDEC ;CONVERT TO DECIMAL
0B70 47             MOV  B, A     ;STORE IN REG B
0B71 E6F0           ANI     0F0H  ;MASK OFF LS DIGIT
0B73 0F             RRC
0B74 0F             RRC
0B75 0F             RRC
0B76 0F             RRC
0B77 CDBC08         CALL    CONAS ;CONVERT TO ASCII
0B7A 2A1220         LHLD    POINT
0B7D 77             MOV  M, A     ;LOAD PRINT STACK
0B7E CD320A         CALL    PRINT ;PRINT MS DIGIT
0B81 3E2E           MVI  A, '.'
0B83 23             INX     H
0B84 77             MOV  M, A     ;LOAD DECIMAL POINT
0B85 CD320A         CALL    PRINT ;PRINT DECIMAL POINT
```

```
0B88 78                 MOV  A, B     ;LOAD NUMBER AGAIN
0B89 E60F               ANI     0FH   ;MASK OFF MS DIGIT
0B8B CDBC08             CALL    CONAS ;CONVERT TO ASCII
0B8E 23                 INX     H
0B8F 77                 MOV  M, A     ;LOAD PRINT STACK
0B90 CD320A             CALL    PRINT ;PRINT LS DIGIT
0B93 CD9D0A             CALL    PRIMM
0B96 20564F4C           DB      ' VOLTS'
0B9A 5453
0B9C 00                 DB      00H
0B9D C33901             JMP     KEYB  ;RETURN
                ;*******************************************
                ;*                                         *
                ;*           HARDWARE TIMER DELAYS         *
                ;*                                         *
                ;*******************************************
                ;
                ;SUBROUTINE TO GENERATE 20 MSEC DELAY
                ;STEPPER CONTROLS ARE ALSO OUTPUT
0BA0 3E00       DEL20:  MVI  A, 00H
0BA2 D304               OUT     STPER ;CLEAR FLIP FLOP
0BA4 3E40               MVI  A, 40H
0BA6 B6                 ORA     M     ;LOAD STEPPER CONTROLS
0BA7 D304               OUT     STPER
                ;  COUNTER NUMBER 1 IS USED
0BA9 3E78               MVI  A, 78H
0BAB D300               OUT     MODEA ;SET COUNTER MODE
0BAD 3E88               MVI  A, 88H
0BAF D302               OUT     TIMA1 ;LOAD LOW ORDER COUNT
0BB1 3E13               MVI  A, 13H
0BB3 D302               OUT     TIMA1 ;LOAD HIGH ORDER COUNT
                ;  WAIT FOR TIMER TO FINISH
0BB5 1          +CHKD:  READ    TEST  ;READ TIMER
0BB5 1 D5       +       PUSH    D     ;SAVE DE REGISTERS
0BB6 1 1608     +       MVI  D, TEST  ;STORE SWITCH NUMBER IN D
0BB8 1 DB06     +       IN      SWCH  ;READ THE SWITCH INPUTS
0BBA 1 1F       +NO:    RAR           ;ROTATE D0 INTO CARRY
0BBB 1 15       +       DCR  D        ;DECREMENT COUNTER
0BBC 1 C2BA0B +         JNZ     NO    ;NOT CORRECT SWITCH YET
0BBF 1 D1       +       POP     D     ;RESTORE DE REGISTERS
0BC0 DAB50B             JC      CHKD  ;TIMER NOT FINISHED
0BC3 C9                 RET
                ;*******************************************
                ;*                                         *
                ;*         CB RADIO CONTROL PROGRAMS       *
                ;*                                         *
                ;*******************************************
                ;
                ;SUBROUTINE TO SET THE CHANNELS TO BE SCANNED
                ;  NDAT IS SET FOR THE CASSETTE ROUTINE
                ;  CHANNEL NUMBERS ARE LOADED INTO THE BUFFER
                ;  CR AND EOF ARE LOADED INTO THE BUFFER
                ;SPACE CHARACTERS ARE ALSO LOADED INTO THE
                ;BUFFER BETWEEN DATA SETS
```

```
                  ;CHN? CONTAINS THE CHANNEL CODES
                  ;CH?  IS THE POSTION IN THE PRINT BUFFER
                  ;      RESERVED FOR THE CHANNEL NUMBERS
                  ;CNUM WILL BE LOADED WITH THE NUMBER
                  ;      OF ACTIVE CHANNELS
0BC4 013620       SAC:   LXI  B, CHN1
0BC7 116921              LXI  D, CH1
0BCA AF                  XRA  A
0BCB 321B21              STA      CNUM  ;CLEAR CHANNEL NUMBER
0BCE CDC701              CALL     CHKEY ;READ KEYBOARD
0BD1 FE20                CPI      ' '   ;IS CHARACTER A SPACE
0BD3 C29E09              JNZ      ERROR ;NOT A SPACE
                  ;  SPACE HAS BEEN RECEIVED,
                  ;  RECEIVE THE CHANNEL NUMBERS.
0BD6 CD880C       GCH:   CALL     GCHAN ;GET CHANNEL
0BD9 02                  STAX B         ;STORE CHANNEL CODE
                  ;  INCREMENT NUMBER OF CHANNELS
0BDA 3A1B21              LDA      CNUM
0BDD 3C                  INR  A
0BDE 321B21              STA      CNUM  ;INCREMENT CHANNEL COUNTER
                  ;  LOAD CHANNEL NUMBER (ASCII) INTO
                  ;  THE PRINT BUFFER
0BE1 3AC121              LDA      CH5
0BE4 12                  STAX D         ;STORE MS DIGIT
0BE5 13                  INX  D
0BE6 3AC221              LDA      CH5+1
0BE9 12                  STAX D         ;LOAD LS DIGIT
                  ;  POINT DE AT THE NEXT CHANNEL DATA
                  ;  LOCATION IN THE PRINT BUFFER.
0BEA CD8D09              CALL     INDE  ;INCREMENT DE 21 TIMES
                  ;  5 CHANNELS STORED YET?
                  ;  NEGATE THE ADDRESS OF THE FIFTH CHANNEL CODE
0BED 213A20              LXI  H, CHN5
0BF0 7C                  MOV  A, H
0BF1 2F                  CMA
0BF2 67                  MOV  H, A
0BF3 7D                  MOV  A, L
0BF4 2F                  CMA
0BF5 6F                  MOV  L, A
0BF6 23                  INX  H         ;NEGATE CHN5
                  ;  SUBTRACT THE ADDRESS OF THE FIFTH CHANNEL
                  ;  CODE FROM THE ADDRESS OF THE PRESENT
                  ;  CHANNEL CODE.
0BF7 09                  DAD  B         ;POINTER-CHN5
                  ;  CHECK THE RESULT OF THE SUBTRACTION
0BF8 7C                  MOV  A, H
0BF9 A7                  ANA  A         ;SET FLAGS
0BFA C2050C              JNZ      CON1
0BFD 7D                  MOV  A, L
0BFE A7                  ANA  A         ;SET FLAGS
0BFF C2050C              JNZ      CON1
                  ;  ALL 5 CHANNELS ARE SET
0C02 C32A0C              JMP      SEPP   ;RETURN
                  ;  NOT ALL FIVE CHANNELS HAVE BEEN SET,
                  ;  POINT BC AT THE NEXT CHANNEL CODE LOCATION
0C05 03           CON1:  INX  B          ;INCREMENT POINTER
                  ;  RECEIVE THE NEXT CHARACTER FROM THE KEYBOARD
0C06 CDC701              CALL    CHKEY ;READ KEYBOARD
0C09 FE2C                CPI     ','   ;IS CHARACTER A COMMA
0C0B CAD60B              JZ      GCH   ;READ ANOTHER NUMBER
                  ;  A COMMA WAS NOT RECEIVED, CHECK TO SEE
                  ;  IF IT WAS A PERIOD.
0C0E FE2E                CPI     '.'   ;IS CHARACTER A PERIOD
0C10 C29E09              JNZ     ERROR
                  ;  A PERIOD WAS RECEIVED, THEREFORE STORE A
                  ;  NULL CHARACTER FOR THE CHANNEL CODE.
0C13 AF           CLRS:  XRA  A          ;CLEAR ACCUMULATOR
0C14 02                  STAX B          ;STORE A NULL CHARACTER
                  ;  ALL 5 CHANNELS STORED YET?
                  ;  NEGATE THE ADDRESS OF THE FIFTH CHANNEL CODE
0C15 213A20              LXI  H, CHN5
0C18 7C                  MOV  A, H
0C19 2F                  CMA
0C1A 67                  MOV  H, A
0C1B 7D                  MOV  A, L
0C1C 2F                  CMA
0C1D 6F                  MOV  L, A
0C1E 23                  INX  H          ;NEGATE CHN5
                  ;  SUBTRACT THE ADDRESS OF THE FIFTH CHANNEL
                  ;  CODE FROM THE ADDRESS OF THE PRESENT
                  ;  CHANNEL CODE.
0C1F 09                  DAD  B          ;POINTER-CHN5
                  ;  CHECK THE RESULT OF THE SUBTRACTION
0C20 7C                  MOV  A, H
0C21 A7                  ANA  A          ;SET FLAGS
0C22 C26E0C              JNZ     CON2
0C25 7D                  MOV  A, L
0C26 A7                  ANA  A          ;SET FLAGS
0C27 C26E0C              JNZ     CON2
                  ;  ALL 5 CHANNELS ARE STORED
                  ;  WITH EITHER CHANNEL CODES OR NULL
                  ;  CHARACTERS. CALCULATE THE VALUE FOR NDAT
0C2A 3A1B21       SEPP:  LDA     CNUM  ;LOAD COUNTER
0C2D 47                  MOV  B, A
0C2E 0E2C                MVI  C, 44    ;LOAD CONSTANT
                  ;  MULTIPLY THE NUMBER OF CHANNELS BY 44
                  ;  AND STORE IN NDAT
0C30 CD4508              CALL    MULT8
0C33 60                  MOV  H, B
0C34 69                  MOV  L, C
0C35 222120              SHLD    NDAT  ;SET NDAT
                  ;  INITIALIZE PRINTER BUFFER
                  ;  POINT DE AT THE CARRIAGE CONTROL LOCATION
                  ;  IN THE PRINT BUFFER
0C38 117721              LXI  D, FLG1+3
0C3B 216821              LXI  H, TM1+6
0C3E 060D                MVI  B, 0DH   ;LOAD CARR RETURN
0C40 3A1B21              LDA  CNUM     ;LOAD COUNTER
```

```
0C43 4F               MOV  C, A
                 ;  REGISTER C CONTAINS THE NUMBER OF CHANNELS
                 ;  REGISTER B CONTAINS A CARRIAGE RETURN
0C44 E5          AGN:   PUSH H
0C45 3E20               MVI  A, 20H
                 ;  PUT A SPACE AFTER THE TIME LOCATION IN
                 ;  THE PRINT BUFFER.
0C47 77                 MOV  M, A      ;STORE A SPACE
                 ;  POINT HL AT THE LOCATION AFTER THE
                 ;  CHANNEL NUMBER
0C48 23                 INX  H
0C49 23                 INX  H
0C4A 23                 INX  H
                 ;  PUT A SPACE AFTER THE CHANNEL NUMBER
0C4B 77                 MOV  M, A      ;STORE A SPACE
                 ;  POINT HL AT THE LOCATION AFTER
                 ;  THE X COORDINATE
0C4C 23                 INX  H
0C4D 23                 INX  H
0C4E 23                 INX  H
0C4F 23                 INX  H
                 ;  PUT A SPACE AFTER THE X COORDINATE
0C50 77                 MOV  M, A      ;STORE A SPACE
                 ;  POINT HL AT THE LOCATION AFTER THE
                 ;  Y COORDINATE
0C51 23                 INX  H
0C52 23                 INX  H
0C53 23                 INX  H
0C54 23                 INX  H
                 ;  PUT A SPACE AFTER THE Y COORDINATE
0C55 77                 MOV  M, A      ;STORE A SPACE
                 ;  RELOAD HL TO POINT AT THE LOCATION
                 ;  AFTER THE TIME IN THE PRINT BUFFER
0C56 E1                 POP  H
                 ;  INCREMENT HL TO POINT AT THE
                 ;  NEXT CHANNEL DATA
0C57 CD8609             CALL    AD21
0C5A 23                 INX  H         ;INCREMENT HL 22 TIMES
                 ;  DECREMENT COUNTER CONTAINING THE NUMBER
                 ;  OF CHANNELS
0C5B 0D                 DCR  C
0C5C CA680C             JZ      LD1C
                 ;  NOT ALL CHANNEL DATA HAS BEEN SET UP
                 ;  IN THE BUFFER
                 ;  PUT A CARRIAGE RETURN AFTER THE PRESENT LINE
0C5F 78                 MOV  A, B
0C60 12                 STAX D
                 ;  POINT DE AT THE CARRIAGE CONTROL LOCATION
                 ;  FOR THE NEXT LINE IN THE BUFFER
0C61 13                 INX  D
0C62 CD8D09             CALL    INDE  ;INCREMENT DE 22 TIMES
                 ;  LOAD THE SPACES FOR THE NEXT DATA LINE
0C65 C3440C             JMP     AGN
                 ;  THIS IS THE LAST LINE OF THE CHANNEL DATA,
                 ;  THEREFORE LOAD AN END OF FILE CHARACTER
0C68 3E1C        LD1C:  MVI  A, 1CH
0C6A 12                 STAX D          ;LOAD END OF FILE
0C6B C33901             JMP      KEYB  ;RETURN
                 ;  A PERIOD HAS BEEN RECEIVED BUT THE
                 ;  REMAINING LOCATIONS HAVE NOT BEEN SET TO
                 ;  ZERO
0C6E 03          CON2:  INX  B          ;INCREMENT POINTER
0C6F C3130C             JMP     CLRS
                 ;
                 ;SUBROUTINE TO SET CB RADIO CHANNEL
                 ;  THE CHANNEL NUMBER IS RECEIVED FROM THE
                 ;  KEYBOARD, CONVERTED TO THE APPROPRIATE
                 ;  CHANNEL CODE, AND SENT TO THE RADIO.
0C72 CDC701      CHN:   CALL     CHKEY ;READ KEYBOARD
0C75 FE20               CPI      ' '   ;IS CHARACTER A SPACE
0C77 C29E09             JNZ      ERROR ;NOT A SPACE
                 ;  A SPACE HAS BEEN RECEIVED
0C7A CD880C             CALL     GCHAN ;GET CHANNEL CODE
0C7D D314               OUT      CHANL ;OUTPUT TO RADIO
                 ;  LOAD CHANNEL NUMBER INTO CHNL SO THAT
                 ;  THE DISPLAY ROUTINES CAN PRINT IT
0C7F 3A3420             LDA      CNL   ;GET CHANNEL NUMBER
0C82 323520             STA      CHNL
0C85 C33901             JMP      KEYB  ;RETURN
                 ;
                 ;SUBROUTINE TO GET CHANNEL CODE
                 ;  CHANNEL NUMBER IN ASCII IS RETURNED IN
                 ;  CH5 AND CH5+1.
                 ;  CHANNEL NUMBER (1 TO 5) IS RETURNED IN CNL
                 ;  CHANNEL CODE IS RETURNED IN REGISTER A
0C88 D5          GCHAN: PUSH D
                 ;  READ MS DIGIT
0C89 CDC701             CALL     CHKEY ;READ MS DIGIT
0C8C 32C121             STA      CH5   ;STORE MS DIGIT
0C8F CD9D08             CALL     CONHX ;CONVERT TO DECIMAL
0C92 DA9E09             JC       ERROR
0C95 07                 RLC
0C96 07                 RLC
0C97 07                 RLC
0C98 07                 RLC
0C99 57                 MOV  D, A       ;STORE IN REG D
                 ;  RECEIVE  LS DIGIT
0C9A CDC701             CALL     CHKEY ;RECEIVE LS DIGIT
0C9D 32C221             STA      CH5+1 ;STORE LS DIGIT
0CA0 CD9D08             CALL     CONHX ;CONVERT TO DECIMAL
0CA3 DA9E09             JC       ERROR
                 ;  D CONTAINS THE MS NIBBLE
                 ;  A CONTAINS THE LS NIBBLE
0CA6 B2                 ORA  D          ;ADD LS DIGIT
0CA7 CD6408             CALL     DECHX ;CONVERT TO HEX
0CAA 323420             STA      CNL   ;STORE
                 ;  REGISTER A CONTAINS THE CHANNEL NUMBER
                 ;  CHECK TO SEE IF THE CHANNEL NUMBER IS VALID
```

```
0CAD 5F              MOV  E, A      ;STORE IN REGISTER E
0CAE FE00            CPI      00H
0CB0 CA9E09          JZ       ERROR ;CHANNEL 0 IS INVALID
0CB3 D629            SUI      41D
0CB5 F29E09          JP       ERROR ;INVALID CHANNEL NUMBER
               ;  THE CHANNEL NUMBER IS VALID, LOOK UP THE
               ;  CHANNEL CODE IN THE TABLE (CHTAB)
0CB8 AF              XRA  A
0CB9 57              MOV  D, A      ;LOAD DE REGISTER
               ;  DE REGISTER CONTAINS THE CHANNEL NUMBER
0CBA 21C00C          LXI  H, CHTAB-1 ;POINT AT START OF TABLE
0CBD 19              DAD  D         ;ADD OFFSET
               ;  HL NOW POINTS AT CORRECT CHANNEL CODE
0CBE 7E              MOV  A, M      ;LOAD CHANNEL CODE
               ;  REGISTER A CONTAINS THE CHANNEL CODE
0CBF D1              POP  D
0CC0 C9              RET
               ;
               ;TABLE FOR CHANNEL CODES FOR CB RADIO
0CC1 3B        CHTAB: DB    00111011B
0CC2 3A              DB    00111010B
0CC3 39              DB    00111001B
0CC4 37              DB    00110111B
0CC5 36              DB    00110110B
0CC6 35              DB    00110101B
0CC7 34              DB    00110100B
0CC8 32              DB    00110010B
0CC9 31              DB    00110001B
0CCA 30              DB    00110000B
0CCB 2F              DB    00101111B
0CCC 2D              DB    00101101B
0CCD 2C              DB    00101100B
0CCE 2B              DB    00101011B
0CCF 2A              DB    00101010B
0CD0 28              DB    00101000B
0CD1 27              DB    00100111B
0CD2 26              DB    00100110B
0CD3 25              DB    00100101B
0CD4 23              DB    00100011B
0CD5 22              DB    00100010B
0CD6 21              DB    00100001B
0CD7 1E              DB    00011110B
0CD8 28              DB    00101000B
0CD9 1F              DB    00011111B
0CDA 1D              DB    00011101B
0CDB 1C              DB    00011100B
0CDC 1B              DB    00011011B
0CDD 1A              DB    00011010B
0CDE 19              DB    00011001B
0CDF 18              DB    00011000B
0CE0 17              DB    00010111B
0CE1 16              DB    00010110B
0CE2 15              DB    00010101B
0CE3 14              DB    00010100B
```

```
0CE4 13                DB    00010011B
0CE5 12                DB    00010010B
0CE6 11                DB    00010001B
0CE7 10                DB    00010000B
0CE8 0F                DB    00001111B
                ;********************************************
                ;*                                          *
                ;*          INTERRUPT SERVICE SUBROUTINE    *
                ;*          FOR RST 5.5 (PRINTER TIMER)     *
                ;* CALLING PROC.- LOAD DATA INTO BUFFER(PREF) *
                ;*    END EACH LINE WITH A CARR. RETURN (0D) *
                ;*    END TEXT WITH AN END OF FILE (1C)     *
                ;*    CALL PTINT                            *
                ;*LOC- STORAGE FOR PROGRAM COUNTER          *
                ;*PRE- POINTER FOR PRINT BUFFER             *
                ;*                                          *
                ;********************************************
0CE9 F5         PTINT: PUSH PSW
0CEA E5                PUSH H
0CEB AF                XRA  A
0CEC D304              OUT      STPER ;CLEAR RST 5.5 FLIP FLOP
0CEE FB                EI             ;ENABLE INTERRUPTS
                ;  LOC CONTAINS THE ADDRESS AT WHICH EXECUTION
                ;  OF THE PROGRAM SHOULD CONTINUE
                ;  IT IS INITIALLY SET TO START
0CEF 2A1421            LHLD     LOC   ;READ LOCATION
0CF2 E9                PCHL           ;JUMP TO LOCATION
                ;  ENABLE RST 5.5
                ;  SET PRINTER ACTIVE FLAG
                ;  WHEN PACT FLAG IS SET, THE SYSTEM KNOWS THAT
                ;  THE PRINTER IS PRINTING. IF THE MODE SWITCH
                ;  IS SET TO CONTROL, THE SYSTEM WILL WAIT FOR
                ;  THE PRINTER TO STOP BEFORE IT PASSES CONTROL
                ;  TO THE KEYBOARD ROUTINE.
0CF3 3EFF       START: MVI  A, 0FFH
0CF5 323022            STA      PACT
0CF8 20                DB       20H   ;RIM
0CF9 E606              ANI      06H
0CFB F608              ORI      08H   ;SET MASK ENABLE
0CFD 30                DB       30H   ;SIM
                ;  INITIALIZE BUFFER POINTER
                ;  THIS IS ONLY EXECUTED THE FIRST TIME
                ;  THROUGH THE PROGRAM
0CFE 21D021            LXI  H, PRBF  ;LOAD POINTER TO BUFFER
0D01 221621            SHLD     PRB
                ;
                ;CARRIAGE RETURN ROUTINE
                ;  THIS ROUTINE WILL MOVE THE PRINTHEAD FROM
                ;  THE HOME POSITION TO THE FIRST PRINT
                ;  POSITION.
                ;  PRCNT IS USED AS A COUNTER TO COUNT THE
                ;  NUMBER OF POSITION LEFT TO MOVE.
0D04 3E12       RTURN: MVI  A, 18D
0D06 321921            STA      PRCNT ;LOAD COUNTER WITH 18
```

```
                ;  LOAD STEPPER CONTROL SIGNALS AND OUTPUT
                ;  THE TO THE MOTOR
0D09 2A1420     ML:     LHLD     PRPOS ;LOAD PRINT POSITION
0D0C 2B                 DCX  H         ;MOVE ONE LOCATION LEFT
0D0D 221420             SHLD     PRPOS
0D10 3E40               MVI  A, 40H    ;ENABLE RST 5.5 FLIP FLOP
0D12 B6                 ORA  M         ;LOAD STEPPER CONTROLS
0D13 D304               OUT      STPER ;OUTPUT TO STEPPER
                ;  COUNTER NUMBER 1 IS USED
0D15 3E78               MVI  A, 78H
0D17 D300               OUT      MODEA ;SET COUNTER MODE
0D19 3E88               MVI  A, 88H
0D1B D302               OUT      TIMA1 ;LOAD LOW ORDER COUNT
0D1D 3E13               MVI  A, 13H
0D1F D302               OUT      TIMA1  ;LOAD HIGH ORDER COUNT
                ;  THE TIMER HAS BEEN LOADED. RETURN TO THE
                ;  INTERRUPTED PROGRAM UNTIL THE TIMER IS DONE.
0D21 CD130E             CALL     LEAVE
                ;  TIMER IS FINISHED, CHECK TO SEE IF THE
                ;  PRINTHEAD IS AT THE FIRST PRINT POSITION.
                ;  20 MSEC. IS OVER
0D24 3A1921             LDA      PRCNT ;LOAD COUNTER
0D27 3D                 DCR  A
0D28 321921             STA      PRCNT ;DECREMENT COUNTER
0D2B C2090D             JNZ      ML    ;NOT DONE YET
                ;  PRINTHEAD IS AT THE FIRST PRINT POSITION
                ;  READ CHARACTER FROM BUFFER
0D2E 2A1621             LHLD     PRB
0D31 7E         NXT1:   MOV  A, M      ;LOAD CHARACTER
0D32 FE0D               CPI      CRET  ;IS CHAR. A CARR. RETURN
0D34 CA950D             JZ       CRET1 ;CARRIAGE RETURN
0D37 FE1C               CPI      1CH   ;IS CHARACTER AN EOF
0D39 CA8D0D             JZ       EOF1  ;END OF FILE
                ;  CHARACTER IS NOT A VALID ASCII CHARACTER,
                ;  THEREFORE PRINT IT.
0D3C D305               OUT      ASCII ;OUTPUT TO CHARACTER GEN.
                ;  COUNTER NUMBER 0 IS USED
0D3E 3E40               MVI  A, 40H
0D40 D304               OUT      STPER ;ENABLE RST 5.5 FLIP FLOP
0D42 3E32               MVI  A, 32H
0D44 D300               OUT      MODEA ;SET COUNTER MODE
0D46 3E2F               MVI  A, 2FH
0D48 D303               OUT      TIMA0 ;LOAD LOW ORDER COUNT
0D4A 3E0D               MVI  A, 0DH
0D4C D303               OUT      TIMA0 ;LOAD HIGH ORDER COUNT
0D4E 3EC0               MVI  A, 0C0H
0D50 D304               OUT      STPER ;TRIGGER PRINT STROBE
                ;  THE TIMER HAS BEEN LOADED, RETURN TO THE
                ;  INTERRUPTED PROGRAM UNTIL THE TIMER IS DONE.
0D52 CD130E             CALL     LEAVE
                ;  CHARACTER HAS BEEN PRINTED
                ;  GENERATE 6 MSEC. DELAY
0D55 3E40               MVI  A, 40H
0D57 D304               OUT      STPER ;ENABLE RST 5.5 FLIP FLOP
```

```
0D59 3E78            MVI  A, 78H
0D5B D300            OUT      MODEA ;SET COUNTER MODE
0D5D 3EDC            MVI  A, 0DCH
0D5F D302            OUT      TIMA1 ;LOAD LOW ORDER COUNT
0D61 3E05            MVI  A, 05H
0D63 D302            OUT      TIMA1 ;LOAD HIGH ORDER COUNT
                ;  THE TIMER HAS BEEN LOADED, RETURN TO THE
                ;  INTERRUPTED PROGRAM UNTIL THE TIMER IS DONE.
0D65 CD130E          CALL     LEAVE
                ;  TIMER IS FINISHED
                ;  MOVE PRINTHEAD TO THE RIGHT
0D68 2A1420          LHLD     PRPOS ;LOAD PRINT POSITION
0D6B 23              INX  H
0D6C 221420          SHLD     PRPOS ;MOVE TO THE RIGHT
0D6F 3E40            MVI  A, 40H
0D71 B6              ORA  M           ;LOAD STEPPER CONTROLS
0D72 D304            OUT      STPER ;OUTPUT TO STEPPER
0D74 3E78            MVI  A, 78H
0D76 D300            OUT      MODEA ;SET COUNTER MODE
0D78 3E88            MVI  A, 88H
0D7A D302            OUT      TIMA1 ;LOAD LOW ORDER COUNT
0D7C 3E13            MVI  A, 13H
0D7E D302            OUT      TIMA1 ;LOAD HIGH ORDER COUNT
                ;  THE TIMER HAS BEEN LOADED, RETURN TO THE
                ;  INTERRUPTED PROGRAM UNTIL THE TIMER IS DONE.
0D80 CD130E          CALL     LEAVE
                ;  20 MSEC. IS OVER
                ;  INCREMENT POINTER TO POINT AT THE NEXT
                ;  CHARATER IN THE PRINT BUFFER
0D83 2A1621          LHLD     PRB
0D86 23              INX  H
0D87 221621          SHLD     PRB   ;INCREMENT POINTER
0D8A C3310D          JMP      NXT1  ;PRINT NEXT CHARACTER
                ;
                ;END OF FILE
                ;  SET EOFF, THIS INDICATES THAT NO MORE
                ;  DATA IS IN THE BUFFER
0D8D 3EFF       EOF1:  MVI  A, 0FFH
0D8F 321821          STA      EOFF  ;SET EOF FLAG
0D92 C3A00D          JMP      HOM1
                ;
                ;CARRIAGE RETURN
                ;  CLEAR EOFF, THIS INDICATES THAT THERE IS
                ;  MORE DATA IN THE BUFFER
0D95 AF         CRET1: XRA  A
0D96 321821          STA      EOFF  ;CLEAR EOF FLAG
                ;  POINT PRB AT THE NEXT CHARACTER TO BE
                ;  PRINTED. IT WILL BE ON THE NEXT LINE.
0D99 2A1621          LHLD     PRB
0D9C 23              INX  H
0D9D 221621          SHLD     PRB   ;INCREMENT POINTER
                ;  MOVE THE PRINTHEAD ONE POSTION TO THE RIGHT
0DA0 2A1420     HOM1:  LHLD     PRPOS ;LOAD PRINTER POSITION
0DA3 23              INX  H
```

```
0DA4 221420           SHLD   PRPOS ;MOVE ONE LOCATION LEFT
0DA7 3E40             MVI  A, 40H   ;ENABLE RST 5.5 FLIP FLOP
0DA9 B6               ORA  M        ;LOAD STEPPER CONTROLS
0DAA D304             OUT    STPER ;OUTPUT TO STEPPER
               ;  COUNTER NUMBER 1 IS USED
0DAC 3E78             MVI  A, 78H
0DAE D300             OUT    MODEA ;SET COUNTER MODE
0DB0 3E88             MVI  A, 88H
0DB2 D302             OUT    TIMA1 ;LOAD LOW ORDER COUNT
0DB4 3E13             MVI  A, 13H
0DB6 D302             OUT    TIMA1 ;LOAD HIGH ORDER COUNT
               ;  THE TIMER HAS BEEN LOADED, RETURN TO THE
               ;  INTERRUPTED PROGRAM UNTIL THE TIMER IS DONE
0DB8 CD130E           CALL   LEAVE
               ;  20 MSEC. IS OVER
  1            +      READ   EOT   ;CHECK EOT SWITCH
0DBB 1 D5      +      PUSH   D     ;SAVE DE REGISTERS
0DBC 1 1601    +      MVI  D, EOT  ;STORE SWITCH NUMBER IN D
0DBE 1 DB06    +      IN     SWCH  ;READ THE SWITCH INPUTS
0DC0 1 1F      +NO:   RAR          ;ROTATE D0 INTO CARRY
0DC1 1 15      +      DCR  D       ;DECREMENT COUNTER
0DC2 1 C2C00D  +      JNZ    NO    ;NOT CORRECT SWITCH YET
0DC5 1 D1      +      POP    D     ;RESTORE DE REGISTERS
0DC6 DAA00D           JC     HOM1  ;NOT DONE YET
               ;  CARRIAGE IS AT END OF TRAVEL
               ;  MOVE LEFT TWICE
0DC9 3E02             MVI  A, 02D
0DCB 321921           STA    PRCNT ;LOAD COUNTER WITH 2
               ;  MOVE PRINTHEAD ONE POSITION TO THE LEFT
0DCE 2A1420    ML1:   LHLD   PRPOS ;LOAD PRINTER POSITION
0DD1 2B               DCX  H
0DD2 221420           SHLD   PRPOS ;MOVE LEFT
0DD5 3E40             MVI  A, 40H   ;ENABLE RST 5.5 FLIP FLOP
0DD7 B6               ORA  M        ;LOAD STEPPER CONTROLS
0DD8 D304             OUT    STPER ;OUTPUT TO STEPPER
               ;  COUNTER NUMBER 1 IS USED
0DDA 3E78             MVI  A, 78H
0DDC D300             OUT    MODEA ;SET COUNTER MODE
0DDE 3E88             MVI  A, 88H
0DE0 D302             OUT    TIMA1 ;LOAD LOW ORDER COUNT
0DE2 3E13             MVI  A, 13H
0DE4 D302             OUT    TIMA1 ;LOAD HIGH ORDER COUNT
               ;  THE TIMER HAS BEEN LOADED, RETURN TO THE
               ;  INTERRUPTED PROGRAM UNTIL THE TIMER IS DONE
0DE6 CD130E           CALL   LEAVE
               ;  20 MSEC. IS OVER
               ;  CHECK TO SEE IF THE PRINTHEAD SHOULD
               ;  BE MOVED AGAIN.
0DE9 3A1921           LDA    PRCNT ;LOAD COUNTER
0DEC 3D               DCR  A
0DED 321921           STA    PRCNT ;DECREMENT COUNTER
0DF0 C2CE0D           JNZ    ML1   ;NOT DONE YET
               ;  RESET PRINTER POSITION POINTER
0DF3 21C609           LXI  H, STEP+18D
```

```
0DF6 221420           SHLD    PRPOS ;INITIALIZE PRPOS
              ;  CHECK TO SEE IF THE PRINTING IS FINISHED.
0DF9 3A1821           LDA     EOFF  ;READ EOF FLAG
0DFC A7               ANA  A        ;SET FLAGS
0DFD CA040D           JZ      RTURN ;CARRIAGE RETURN
              ;  END OF FILE HAS BEEN RECEIVED
              ;  EXIT FROM INTERRUPT ROUTINE
              ;  LOAD LOC WITH THE ADDRESS OF START.
              ;  THIS WILL ENSURE THAT THE NEXT TIME THIS
              ;  ROUTINE IS CALLED, EXECUTION WILL BEGIN
              ;  AT THE START OF THE ROUTINE.
0E00 21F30C           LXI  H, START
0E03 221421           SHLD    LOC   ;RESET RETURN LOCATION
              ;  MASK RST 5.5
              ;  THE TIMER CAN THEREFORE NOT INTERRUPT THE
              ;  SYSTEM. THE INTERRUPT WILL BE REENABLED
              ;  THE NEXT TIME THE ROUTINE IS CALLED.
0E06 20               DB      20H   ;RIM
0E07 E607             ANI     07H
0E09 F609             ORI     09H   ;MASK RST 5.5
0E0B 30               DB      30H   ;SIM
              ;  CLEAR PRINTER ACTIVE FLAG
              ;  THE PRINTER IS NO LONGER PRINTING
0E0C AF               XRA  A
0E0D 323022           STA     PACT
0E10 E1               POP  H
0E11 F1               POP  PSW
0E12 C9               RET
              ;  ROUTINE TO LEAVE INTERRUPT ROUTINE AND SET
              ;  RETURN ADDRESS IN LOC
              ;  UPON RECEIVING AN INTERRUPT, PROGRAM
              ;  EXECUTION WILL CONTINUE WITH THE
              ;  INSTRUCTION IMMEDIATELY AFTER THE
              ;  CALL LEAVE INSTRUCTION.
0E13 E1       LEAVE: POP  H         ;LOAD HL WITH RETURN ADD.
0E14 221421           SHLD    LOC   ;STORE ADDRESS IN LOC
0E17 E1               POP  H
0E18 F1               POP  PSW
0E19 C9               RET
              ;**********************************************
              ;*                                            *
              ;*    INTERRUPT DRIVEN PRINTER SUBROUTINES    *
              ;*                                            *
              ;**********************************************
              ;PROF IS READ BY THE SCAN ROUINES TO DETERMINE
              ;     WHETHER OR NOT THE DATA SHOULD BE PRINTED
              ;ACLOC IS READ BY THE SCAN ROUTINES TO
              ;      DETERMINE WHETHER TO PRINT THE LOCATIONS
              ;      OR THE FLAGS. WHEN ACLOC IS SET THE
              ;      FLAGS ARE PRINTED. WHEN ACLOC IS RESET
              ;      THE POSITIONS ARE PRINTED.
              ;
              ;  SUBROUTINE TO ENABLE PRINTER
              ;  AND SET THE PRINT INTERVAL
```

```
0E1A 3EFF      PRI:    MVI  A, OFFH
0E1C 322F22            STA      PROF  ;SET PRINTER FLAG
0E1F CDDB08            CALL     DECKY
0E22 323122            STA      PINT
0E25 323222            STA      PINT1 ;STORE PRINT INTERVAL
0E28 C33901            JMP      KEYB  ;RETURN
               ;
               ;  SUBROUTINE TO DISABLE PRINTER
0E2B AF        POP:    XRA  A
0E2C 322F22            STA      PROF  ;CLEAR PRINTER FLAG
0E2F C33901            JMP      KEYB  ;RETURN
               ;
               ;SUBROUTINE TO SET ACTIVITY PRINT FLAG
0E32 3EFF      FLG:    MVI  A, OFFH
0E34 321A21            STA      ACLOC ;SET FLAG
0E37 C33901            JMP      KEYB  ;RETURN
               ;
               ;SUBROUTINE TO CLEAR ACTIVITY PRINT FLAG
0E3A AF        POS:    XRA  A
0E3B 321A21            STA      ACLOC ;CLEAR FLAG
0E3E C33901            JMP      KEYB  ;RETURN
               ;*********************************************
               ;*                                           *
               ;*        REAL TIME CLOCK SUBROUTINES        *
               ;*                                           *
               ;*********************************************
               ;
               ;SUBROUTINE TO SET THE REAL TIME CLOCK
               ;TIME IS READ FROM THE KEYBOARD IN THE FORMAT
               ;            HH:MM
               ;WHERE HH IS THE TIME IN HOURS
               ;      MM IS THE TIME IN MINUTES
               ;THE CLOCK IS STARTED WHEN A SPACE
               ;IS RECEIVED.
0E41 E5        TST:    PUSH     H
               ;  THE INTERRUPTS ARE DISABLED SO THAT THE
               ;  CLOCK WILL NOT BE INCREMENTED.
0E42 20                DB       20H   ;RIM
0E43 E607              ANI      07H
0E45 F60A              ORI      0AH   ;MASK RST 6.5
0E47 30                DB       30H   ;SIM
0E48 AF                XRA      A     ;CLEAR ACCUMULATOR
               ;  CLEAR THE SECONDS COUNTER
0E49 211720            LXI  H, SEC0
0E4C 77                MOV  M, A      ;CLEAR SEC0
0E4D 23                INX      H
0E4E 77                MOV  M, A      ;CLEAR SEC1
               ;  WAIT FOR SPACE FROM KEYBOARD
0E4F 211C20            LXI  H, HR1
0E52 CDC701            CALL     CHKEY ;READ KEYBOARD
0E55 FE20              CPI      ' '   ;IS CHARACTER A SPACE
0E57 C29E09            JNZ      ERROR ;NOT A SPACE
               ;  A SPACE HAS BEEN RECEIVED, RECEIVE THE
               ;  MS BYTE OF THE HOUR
```

```
0E5A CDC701          CALL    CHKEY ;READ KEYBOARD
0E5D CD9D08          CALL    CONHX ;CONVERT TO HEX
0E60 DA9E09          JC      ERROR ;ERROR EXIT
                ;  LOAD MS DIGIT OF HOURS INTO THE COUNTER
0E63 77              MOV  M, A      ;LOAD HR1
0E64 2B              DCX     H
                ;  RECEIVE LS DIGIT OF THE HOUR
0E65 CDC701          CALL    CHKEY ;READ KEYBOARD
0E68 CD9D08          CALL    CONHX ;CONVERT TO HEX
0E6B DA9E09          JC      ERROR ;ERROR EXIT
                ;  LOAD LS DIGIT FO HOURS INTO THE CLOCK
0E6E 77              MOV  M, A      ;LOAD HR0
0E6F 2B              DCX     H
                ;  WAIT FOR A COLON
0E70 CDC701          CALL    CHKEY ;READ KEYBOARD
0E73 FE3A            CPI     ':'   ;IS CHARACTER A COLON
0E75 C29E09          JNZ     ERROR ;NOT A COLON
                ;  COLON HAS BEEN RECEIVED,
                ;  RECEIVE THE MS DIGIT OF THE MINUTES
0E78 CDC701          CALL    CHKEY ;READ KEYBOARD
0E7B CD9D08          CALL    CONHX ;CONVERT TO HEX
0E7E DA9E09          JC      ERROR ;ERROR EXIT
                ;  LOAD MS DIGIT OF THE MINUTES
0E81 77              MOV  M, A      ;LOAD MIN1
0E82 2B              DCX     H
                ;  RECEIVE LS DIGIT OF THE MINUTES
0E83 CDC701          CALL    CHKEY ;READ KEYBOARD
0E86 CD9D08          CALL    CONHX ;CONVERT TO HEX
0E89 DA9E09          JC      ERROR ;ERROR EXIT
                ;  LOAD THE LS DIGIT OF THE MINUTES
0E8C 77              MOV  M, A      ;LOAD MIN0
                ;  WAIT FOR A SPACE FROM THE KEYBOARD
0E8D CDC701     WAIT1: CALL  CHKEY ;READ KEYBOARD
0E90 FE20            CPI     ' '   ;IS CHARACTER A SPACE
0E92 C28D0E          JNZ     WAIT1 ;WAIT FOR A SPACE
                ;  A SPACE HAS BEEN RECEIVED
                ;  UNMASK RST 6.5
0E95 20              DB      20H   ;RIM
0E96 E605            ANI     05H
0E98 F608            ORI     08H   ;SET MASK ENABLE
0E9A 30              DB      30H   ;SIM
0E9B E1              POP     H
0E9C C33901          JMP     KEYB
                ;
                ;SUBROUTINE TO READ THE REAL TIME CLOCK
0E9F CDA50E     TIM:   CALL    TIME
0EA2 C33901            JMP     KEYB  ;RETURN
0EA5 C5         TIME:  PUSH    B
0EA6 D5                PUSH    D
0EA7 E5                PUSH    H
                ;  DISABLE THE INTERRUPT UNTIL
                ;  THE CLOCK HAS BEEN READ
0EA8 20                DB      20H   ;RIM
0EA9 E607              ANI     07H
```

```
0EAB F60A           ORI     0AH    ;MASK RST 6.5
0EAD 30             DB      30H    ;SIM
0EAE 111120         LXI  D, CHR18
0EB1 211720         LXI  H, SEC0
0EB4 0606           MVI  B, 06H
             ;  PUT TIME IN TEMPORARY STORAGE
0EB6 7E      HOLD:  MOV  A, M
0EB7 12             STAX    D
0EB8 23             INX     H
0EB9 1B             DCX     D
0EBA 05             DCR     B
0EBB C2B60E         JNZ     HOLD
             ;  NOW THAT THE CLOCK HAS BEEN READ
             ;  UNMASK RST 6.5
0EBE 20             DB      20H    ;RIM
0EBF E605           ANI     05H
0EC1 F608           ORI     08H    ;SET MASK ENABLE
0EC3 30             DB      30H;   ;SIM
             ;  PRINT TIME
             ;  THE TIME HAS BEEN STORED IN THE TOP
             ;  OF THE PRINT STACK
             ;  DE POINTS AT THE TIME
0EC4 110C20         LXI  D, CHR18-5
0EC7 CD9D0A         CALL    PRIMM
0ECA 45204953       DB      'E IS '
0ECE 20
0ECF 00             DB      00H
0ED0 210920         LXI  H, CHR10
             ;  HL POINTS AT THE PRINT STACK
             ;  LOAD MS DIGIT OF HOURS
0ED3 1A             LDAX    D
0ED4 CDBC08         CALL    CONAS ;CONVERT TO ASCII
0ED7 DA9E09         JC      ERROR ;ERROR EXIT
             ;  PUT ON PRINT STACK
0EDA 77             MOV  M, A
0EDB 23             INX     H
0EDC 13             INX     D
0EDD CD320A         CALL    PRINT ;PRINT HR1
             ;  LOAD LS DIGIT OF HOURS
0EE0 1A             LDAX    D
0EE1 CDBC08         CALL    CONAS ;CONVERT TO ASCII
0EE4 DA9E09         JC      ERROR ;ERROR EXIT
             ;  PUT ONTO PRINT STACK
0EE7 77             MOV  M, A
0EE8 23             INX     H
0EE9 13             INX     D
0EEA CD320A         CALL    PRINT ;PRINT HR0
             ;  PUT A COLON ONTO PRINT STACK
0EED 3E3A           MVI  A, ':'
0EEF 77             MOV  M, A
0EF0 23             INX     H
0EF1 CD320A         CALL    PRINT ;PRINT A COLON
             ;  LOAD MS DIGIT OF MINUTES
0EF4 1A             LDAX    D
```

```
0EF5 CDBC08         CALL    CONAS ;CONVERT TO ASCII
0EF8 DA9E09         JC      ERROR ;ERROR EXIT
             ;  PUT ONTO PRINT STACK
0EFB 77             MOV  M, A
0EFC 23             INX     H
0EFD 13             INX     D
0EFE CD320A         CALL    PRINT ;PRINT MIN1
             ;  LOAD LS DIGIT OF MINUTES
0F01 1A             LDAX    D
0F02 CDBC08         CALL    CONAS ;CONVERT TO ASCII
0F05 DA9E09         JC      ERROR ;ERROR EXIT
             ;  PUT ONTO PRINT STACK
0F08 77             MOV  M, A
0F09 23             INX     H
0F0A 13             INX     D
0F0B CD320A         CALL    PRINT ;PRINT MIN0
             ;  PUT A COLON ONTO THE PRINT STACK
0F0E 3E3A           MVI  A, ':'
0F10 77             MOV  M, A
0F11 23             INX     H
0F12 CD320A         CALL    PRINT ;PRINT A COLON
             ;  LOAD MS DIGIT OF SECONDS
0F15 1A             LDAX    D
0F16 CDBC08         CALL    CONAS ;CONVERT TO ASCII
0F19 DA9E09         JC      ERROR ;ERROR EXIT
             ;  PUT ONTO PRINT STACK
0F1C 77             MOV  M, A
0F1D 23             INX     H
0F1E 13             INX     D
0F1F CD320A         CALL    PRINT ;PRINT SEC1
             ;  LOAD LS DIGIT OF SECONDS
0F22 1A             LDAX    D
0F23 CDBC08         CALL    CONAS ;CONVERT TO ASCII
0F26 DA9E09         JC      ERROR ;ERROR EXIT
             ;  PUT ONTO PRINT STACK
0F29 77             MOV  M, A
0F2A CD320A         CALL    PRINT ;PRINT SEC0
0F2D E1             POP     H
0F2E D1             POP     D
0F2F C1             POP     B
0F30 C9             RET
             ;**********************************************
             ;*                                            *
             ;*              SCAN SUBROUTINES              *
             ;*                                            *
             ;**********************************************
             ;DECKY IS A SUBROUTINE WHICH RECEIVES TWO
             ;DECIMAL DIGITS FROM THE KEYBOARD AND
             ;RETURNS THE CORRESPONDING HEXADECIMAL
             ;NUMBER IN THE KEYBOARD
             ;
             ;SUBROUTINE TO SET THE SCAN INTERVAL
0F31 CDDB08  SCI:   CALL    DECKY
0F34 323322         STA     SCAN1
```

```
0F37 323422          STA     SCAN2
0F3A C33901          JMP     KEYB
                ;
                ;SUBROUTINE TO SET THE CASSETTE STORE INTERVAL
0F3D CDDB08     CSI:  CALL    DECKY
0F40 323522           STA     CASIP
0F43 323622           STA     CASI1
0F46 323722           STA     CASI2
0F49 C33901           JMP     KEYB
                ;
                ;SUBROUTINE TO SET THE MULTIPLIER CONSTANT
0F4C CDDB08     MLC:  CALL    DECKY
0F4F 323F22           STA     CMULT
0F52 C33901           JMP     KEYB
                ;
                ;SUBROUTINE TO SET THE TEST ANTENNA COMP. VALUE
0F55 CDDB08     TAN:  CALL    DECKY
0F58 321221           STA     TANCP
0F5B C33901           JMP     KEYB
                ;
                ;SUBROUTINE TO SET THE DISTANCE AT WHICH
                ;  THE SCAN RATE WILL INCREASE
0F5E CDDB08     DIS:  CALL    DECKY
0F61 321C21           STA     DISTC
0F64 C33901           JMP     KEYB
                ;
                ;SET PERCENTAGE FOR ACTIVITY CALCULATIONS
                ;THE PERCENTAGE IS STORED AS 100-INPUT
0F67 CDDB08     PCT:  CALL    DECKY
0F6A 47               MOV  B, A        ;STORE IN REGISTER B
0F6B 3E64             MVI  A, 100
0F6D 90               SUB  B           ;100-INPUT
0F6E 321D21           STA     ACPCT
0F71 C33901           JMP     KEYB  ;RETURN
                ;
                ;SUBROUTINE TO CLEAR POPULATION MATRIX
0F74 CD7A0F     CLR:  CALL    SCLR
0F77 C33901           JMP     KEYB
0F7A 11560A     SCLR: LXI  D, 2646  ;LOAD COUNTER
0F7D 21A025           LXI  H, POPM  ;LOAD POINTER
0F80 3600       CLR5: MVI  M, 0     ;CLEAR DATA
0F82 23               INX  H        ;INCREMENT POINTER
0F83 1B               DCX  D        ;DECREMENT COUNTER
                ;  CHECK COUNTER TO SEE IF ZERO
0F84 7B               MOV  A, E
0F85 A7               ANA  A        ;SET FLAGS
0F86 C2800F           JNZ     CLR5  ;NOT DONE YET
0F89 7A               MOV  A, D
0F8A A7               ANA  A        ;SET FLAGS
0F8B C2800F           JNZ     CLR5  ;NOT DONE YET
                ;  ALL LOCATIONS HAVE BEEN CLEARED
0F8E C9               RET
                ;
                ;SUBROUTINE TO DISPLAY POPULATION MATRIX
```

```
                  ;  TRANS. NUMBER (1 TO 3) IS ENTERED FROM KEYB.
0F8F CDDB08   PPM:    CALL    DECKY
0F92 21A025           LXI  H, POPM
                  ;  A CONTAINS THE TRANSMITTER NUMBER
0F95 3D       POP1:  DCR  A          ;DECREMENT COUNTER
0F96 CA9F0F           JZ      POP2
                  ;  HL DOES NOT POINT AT THE CORRECT
                  ;  TRANSMITTER DATA
0F99 CD9709           CALL    AD882 ;INCREMENT POINTER
0F9C C3950F           JMP     POP1
                  ;  HL NOW POINTS AT THE CORRECT CHANNEL DATA
                  ;  USE B AS X COUNTER AND C AS Y COUNTER
0F9F AF       POP2:  XRA  A
0FA0 47               MOV  B, A
0FA1 4F               MOV  C, A       ;CLEAR COUNTERS
                  ;  COUNTERS HAVE BEEN CLEARED
0FA2 CD000A   POP3:  CALL    HOME  ;LINE FEED
0FA5 CD260A           CALL    RETN  ;CARR. RETURN
0FA8 CD9D0A           CALL    PRIMM
0FAB 5820             DB    'X '
0FAD 00               DB      00H
                  ;  PRINT THE X COORDINATE
0FAE 78               MOV  A, B       ;LOAD X COUNTER
0FAF CD2108           CALL    PRBYT ;PRINT X COORDINATE
0FB2 CD9D0A           CALL    PRIMM
0FB5 2C205920         DB    ', Y '
0FB9 00               DB      00H
0FBA 79               MOV  A, C
                  ;  PRINT THE Y COORDINATE
0FBB CD2108           CALL    PRBYT ;PRINT Y COORDINATE
0FBE CD9D0A           CALL    PRIMM
0FC1 203D20           DB    ' = '
0FC4 00               DB      00H
                  ;  LOAD THE POPULATION COUNTER
0FC5 56               MOV  D, M
0FC6 23               INX  H
0FC7 5E               MOV  E, M       ;LOAD DATA INTO DE REG.
0FC8 23               INX  H
                  ;  COUNTER HAS BEEN LOADED INTO DE
                  ;  PRT16 IS A SUBROUTINE WHICH WILL
                  ;  CONVERT A 16 BIT BINARY NUMBER INTO A
                  ;  DECIMAL NUMBER AND PRINT THE NUMBER.
0FC9 CDDF0F           CALL    PRT16 ;PRINT DATA
                  ;  INCREMENT THE Y COUNTER
0FCC 0C               INR  C
0FCD 79               MOV  A, C       ;CHECK Y COUNTER
0FCE FE15             CPI     21      ;DONE YET
0FD0 C2A20F           JNZ     POP3  ;NOT DONE YET
                  ;  Y IS DONE, INCREMENT THE X COUNTER
0FD3 0E00             MVI  C, 0       ;RESET Y COUNTER
0FD5 04               INR  B
                  ;  IS X DONE?
0FD6 78               MOV  A, B       ;CHECK X COUNTER
0FD7 FE15             CPI     21
```

```
0FD9 C2A20F           JNZ     POP3  ;NOT DONE YET
                ;  ALL LOCATIONS HAVE BEEN PRINTED
0FDC C33901           JMP     KEYB  ;RETURN
                ;
                ;SUBROUTINE TO PRINT THE DATA IN THE DE REG.
                ;  THE DATA IS PRINTED IN DECIMAL.
                ;  B IS CLEARED IF ZEROS ARE NOT TO BE PRINTED
                ;  B IS SET IF ZEROS SHOULD BE PRINTED
                ;  PRTNB IS A SUBROUTINE WHICH WILL
                ;  PRINT ONE CHARACTER
0FDF D5         PRT16: PUSH D
0FE0 E5                PUSH H
0FE1 C5                PUSH B
0FE2 F5                PUSH PSW
0FE3 0600              MVI  B, 0      ;DO NOT PRINT LEADING 0'S
0FE5 EB                XCHG
0FE6 11F0D8            LXI  D, -10000 ;LOAD FIRST DIVISOR
0FE9 0E00              MVI  C, 0      ;CLEAR COUNTER
0FEB 19         PRTN1: DAD  D         ;SUBTRACT DIVISOR
0FEC 7C                MOV  A, H
0FED A7                ANA  A         ;SET FLAGS
0FEE FAF50F            JM       THOUS ;FIRST DIGIT IS DONE
0FF1 0C                INR  C         ;INCREMENT DIGIT
0FF2 C3EB0F            JMP      PRTN1
0FF5 111027     THOUS: LXI  D, 10000
0FF8 19                DAD  D         ;ADD DIVISOR BACK
0FF9 79                MOV  A, C      ;LOAD DIGIT
0FFA CD4D10            CALL     PRTNB ;PRINT DIGIT
0FFD 1118FC            LXI  D, -1000  ;LOAD DIVISOR
1000 0E00              MVI  C, 0      ;CLEAR COUNTER
1002 19         PRTN2: DAD  D
1003 7C                MOV  A, H
1004 A7                ANA  A         ;SET FLAGS
1005 FA0C10            JM       HUNDR ;SECOND DIGIT IS DONE
1008 0C                INR  C         ;INCREMENT DIGIT
1009 C30210            JMP      PRTN2
100C 11E803     HUNDR: LXI  D, 1000
100F 19                DAD  D         ;ADD DIVISOR BACK
1010 79                MOV  A, C      ;LOAD DIGIT
1011 CD4D10            CALL     PRTNB ;PRINT DIGIT
1014 119CFF            LXI  D, -100   ;LOAD DIVISOR
1017 0E00              MVI  C, 0      ;CLEAR COUNTER
1019 19         PRTN3: DAD  D
101A 7C                MOV  A, H
101B A7                ANA  A         ;SET FLAGS
101C FA2310            JM       TENS  ;THIRD DIGIT IS DONE
101F 0C                INR  C         ;INCREMENT DIGIT
1020 C31910            JMP      PRTN3
1023 116400     TENS:  LXI  D, 100
1026 19                DAD  D         ;ADD DIVISOR BACK
1027 79                MOV  A, C      ;LOAD DIGIT
1028 CD4D10            CALL     PRTNB ;PRINT DIGIT
102B 11F6FF            LXI  D, -10    ;LOAD DIVISOR
102E 0E00              MVI  C, 0      ;CLEAR COUNTER
```

```
1030 19      PRTN4: DAD  D
1031 7C             MOV  A, H
1032 A7             ANA  A       ;SET FLAGS
1033 FA3A10         JM      ONES  ;FOURTH DIGIT IS DONE
1036 0C             INR  C
1037 C33010         JMP     PRTN4
103A 110A00  ONES:  LXI  D, 10
103D 19             DAD  D       ;ADD DIVISOR BACK
103E 79             MOV  A, C    ;LOAD DIGIT
103F CD4D10         CALL    PRTNB ;PRINT DIGIT
1042 7D             MOV  A, L    ;LOAD LAST DIGIT
             ;  SINCE THIS IS THE LAST DIGIT,
             ;  IT MUST BE PRINTED EVEN IF
             ;  IT IS ZERO
1043 06FF           MVI  B, 0FFH  ;PRINT IF ZERO
1045 CD4D10         CALL    PRTNB ;PRINT DIGIT
1048 F1             POP  PSW
1049 C1             POP  B
104A E1             POP  H
104B D1             POP  D
104C C9             RET
             ;
             ;SUBROUTINE TO PRINT ONE DIGIT IN THE ACC.
             ;  REGISTER B IS SET IF ZERO SHOULD BE PRINTED
104D E5      PRTNB: PUSH H
104E F5             PUSH PSW
             ;  CHECK TO SEE IF CHARACTER IS ZERO
104F A7             ANA  A       ;SET FLAGS
1050 C25810         JNZ     PRTN8 ;PRINT CHARACTER
             ;  CHARACTER IS A ZERO, CHECK TO SEE IF
             ;  IT SHOULD BE PRINTED
1053 78             MOV  A, B    ;LOAD FLAG
1054 A7             ANA  A       ;SET FLAGS
1055 CA6610         JZ      PREXT ;DO NOT PRINT ZERO
             ;  PRINT THE CHARACTER
1058 F1      PRTN8: POP  PSW
1059 F5             PUSH PSW      ;RELOAD DIGIT
105A CDBC08         CALL    CONAS ;CONVERT TO ASCII
105D 2A1220         LHLD    POINT
             ;  PUT CHARACTER INTO PRINT STACK
1060 77             MOV  M, A    ;LOAD CHARACTER
1061 CD320A         CALL    PRINT ;PRINT THE CHARACTER
             ;  SINCE A CHARACTER HAS BEEN PRINTED,
             ;  ALL ZEROS SHOULD NOW BE PRINTED.
1064 06FF           MVI  B, 0FFH  ;NO MORE LEADING ZEROS
1066 F1      PREXT: POP  PSW
1067 E1             POP  H
1068 C9             RET
```

```
                ;*******************************************
                ;*                                         *
                ;*      INTERRRUPT SERVICE SUBROUTINE      *
                ;*      FOR RST 7.5 (CASSETTE INTERFACE)   *
                ;*                                         *
                ;*******************************************
1069 F5         COUT:  PUSH PSW
106A E5                PUSH H
106B D5                PUSH D
                ;  CHECK STATUS OF CASSETTE DRIVE
                ;  CRDY IS SET WHEN THE CASSETTE DRIVE
                ;  IS READY TO RECEIVE DATA
                ;  WHEN THE CASSETTE DRIVE IS FIRST
                ;  TURNED ON, IT IS NOT READY TO
                ;  RECEIVE DATA UNTIL IT IS UP TO SPEED
106C 3A2820            LDA     CRDY  ;CHECK CASSETTE
106F A7                ANA  A        ;SET FLAGS
1070 C27A10            JNZ     DOUT  ;READY TO RECEIVE DATA
                ;  NOT READY TO RECEIVE DATA
                ;  THERFORE OUTPUT A DELETE
1073 3E7F              MVI  A, DELET ;LOAD DELETE
1075 D308              OUT     CASET ;OUTPUT DELETE
1077 C3C410            JMP     EXT
                ;  OUTPUT DATA TO CASSETTE DRIVE
                ;  OUTD IS A POINTER WHICH POINTS TO THE
                ;  CHARACTER IN THE CASSETTE BUFFER WHICH
                ;  IS TO BE OUTPUT NEXT
107A 2A2320     DOUT:  LHLD    OUTD
107D 7E                MOV  A, M     ;LOAD DATA
107E D308              OUT     CASET
                ;  POINT OUTD AT NEXT CHARACTER IN BUFFER
1080 23                INX  H
1081 222320            SHLD    OUTD  ;INCREMENT OUTD
                ;  CHECK FOR END OF DATA TRANSFER
                ;  OTEND CONTAINS THE ADDRESS OF THE LAST
                ;  CHARACTER TO BE OUTPUT
                ;  NEGATE OTEND AND STORE IN DE
1084 3A2A20            LDA     OTEND
1087 2F                CMA
1088 5F                MOV  E, A
1089 3A2B20            LDA     OTEND+1
108C 2F                CMA
108D 57                MOV  D, A     ;LOAD OTEND INT DE
108E 13                INX  D        ;NEGATE OTEND
                ;  DE NOW CONTAINS -OTEND
                ;  HL CONTAINS OUTD
108F 19                DAD  D        ;OUTD-OTEND
                ;  CHECK IF ALL CHARACTER HAVE BEEN OUTPUT
1090 7C                MOV  A, H
1091 07                RLC           ;LOAD CARRY BIT
1092 DAC410            JC      EXT   ;TRANSFER NOT COMPLETED
                ;  DATA TRANSFER COMPLETED
                ;  WAIT FOR UART TO EMPTY
1095 DB09       CUART: IN      C8251 ;READ UART STATUS
```

```
1097 E604           ANI  00000100B
1099 CA9510         JZ       CUART ;UART NOT EMPTY
             ;  UART IS EMPTY, DISABLE IT
109C 3E10           MVI  A, 10H
109E D309           OUT      C8251 ;DISABLE TRANSMITTER
             ;  CHECK IF THE CASSETTE TAPE IS FULL.
             ;  CFIN IS SET IF THE CASSETTE IS FULL
             ;  AND THE CASSETTE SYSTEM SHOULD BE RESET
10A0 3A2920         LDA      CFIN
10A3 A7             ANA  A        ;SET FLAGS
10A4 CAB410         JZ       OFF
             ;  A NEW CASSETTE HAS BEEN INSERTED,
             ;  RESET THE CASSETTE SYSTEM
10A7 AF             XRA  A
10A8 322920         STA      CFIN
10AB 2A3120         LHLD     CSCT
10AE 23             INX  H
10AF 222D20         SHLD     CASCT ;RESET TIMER
             ;  TURN ON THE CASSETTE DONE LED AND
             ;  TURN OFF THE CASSETTE DRIVE
10B2 3E04           MVI  A, 04H   ;TURN ON FINISHED LAMP
10B4 D30A    OFF:   OUT      CPORT ;TURN OFF CASSETTE SYSTEM
             ;  CLEAR CASON. CASON IS A FLAG WHICH
             ;  INDICATES WHETHER OR NOT THE CASSETTE
             ;  DRIVE IS TURNED ON.
10B6 AF             XRA  A        ;CLEAR ACCUMULATOR
10B7 322C20         STA      CASON ;CASSETTE OFF
             ;  SINCE IT WILL TAKE TIME FOR THE CASSETTE
             ;  DRIVE TO SLOW DOWN, DECREMENT THE TIMER
10BA 2A2D20         LHLD     CASCT
10BD 2B             DCX  H        ;DECREMENT CASSETTE TIMER
10BE 222D20         SHLD     CASCT ;  FOR TAPE SLOWDOWN TIME
             ;  RESET THE INTERRUPT FLIP FLOP
10C1 3E10           MVI  A, 10H
10C3 30             DB       30H   ;RESET RST 7.5
10C4 D1      EXT:   POP  D
10C5 E1             POP  H
10C6 F1             POP  PSW
10C7 FB             EI
10C8 C9             RET
```

```
                ;*********************************************
                ;*                                           *
                ;*        CASSETTE INTERFACE SUBROUTINES     *
                ;*  CSTRT  CONTAINS THE STARTING ADDRESS     *
                ;*         OF THE BUFFER                     *
                ;*  CEND   CONTAINS THE ENDING ADDRESS       *
                ;*         OF THE BUFFER                     *
                ;*                                           *
                ;*********************************************
                ;DATA TO THE CASSETTE BUFFER IS IN THE
                ;FOLLOWING FORMAT
                ;   <GS>ASCII DATA<DEL><CR>
                ;THE LAST LINE IN THE FILE IS IN THE
                ;FOLLOWINF FORMAT
                ;   <GS>ASCII DATA<DEL><FS><FS>
                ;WHERE <GS> IS A GROUP SEPARATOR (1D)
                ;      <DEL> IS A DELETE CHARACTER (7F)
                ;      <CR> IS A CARRIAGE RETURN (0D)
                ;      <FS> IS A FILE SEPARATOR (1C)
                ;
                ;SUB. TO LOAD THE STARTING ADD. OF THE BUFFER
                ;FOR THE CASSETTE TAPE SYSTEM.
10C9 E5         CST:   PUSH H
10CA CDC701            CALL    CHKEY ;READ KEYBOARD
10CD FE20              CPI     ' '   ;IS CHARACTER A SPACE
10CF C29E09            JNZ     ERROR ;NOT A SPACE
                ;  A SPACE HAS BEEN RECEIVED, RECEIVE
                ;  THE STARTING ADDRESS
10D2 CD4E09            CALL    GTADD ;GET ADDRESS
10D5 221D20            SHLD    CSTRT ;LOAD STARTING ADDRESS
                ;  INITIALIZE THE BUFFER POINTER TO START
                ;  AT THE BEGINNING OF THE BUFFER
10D8 222520            SHLD    BUFER ;INITIALIZE LOADING ADD.
10DB E1                POP  H
10DC C33901            JMP     KEYB  ;RETURN
                ;
                ;SUB. TO LOAD THE ENDING ADDRESS OF THE BUFFER
                ;FOR THE CASSETTE TAPE SYSTEM.
10DF E5         CEN:   PUSH H
10E0 CDC701            CALL    CHKEY ;READ KEYBOARD
10E3 FE20              CPI     ' '   ;IS CHARACTER A SPACE
10E5 C29E09            JNZ     ERROR ;NOT A SPACE
                ;  A SPACE HAS BEEN RECEIVED, RECEIVE
                ;  THE ENDING ADDRESS.
10E8 CD4E09            CALL    GTADD ;GET ADDRESS
10EB 221F20            SHLD    CEND ;LOAD ENDING ADDRESS
10EE E1                POP  H
10EF C33901            JMP     KEYB  ;RETURN
                ;
                ;SUBROUTINE TO TURN ON ALARM
10F2 3E02       ALM:   MVI  A, 02H
10F4 D30A              OUT     CPORT
10F6 AF                XRA  A
10F7 D30A              OUT     CPORT ;TOGGLE ALARM FLIP FLOP
```

```
10F9 C33901          JMP     KEYB  ;RETURN
               ;
               ;SUBROUTINE TO TURN ON THE CASSETTE DRIVE
10FC 3E01      CON:  MVI  A, 01H
10FE D30A            OUT     CPORT ;TURN ON DRIVE
1100 C33901          JMP     KEYB  ;RETURN
               ;
               ;SUBROUTINE TO TURN OFF THE CASSETTE DRIVE
1103 AF        COF:  XRA  A
1104 D30A            OUT     CPORT ;TURN OFF DRIVE
1106 C33901          JMP     KEYB  ;RETURN
               ;
               ;SUBROUTINE TO SET SIZE OF CASSETTE TAPE
1109 CDDB08    CTP:  CALL    DECKY
110C 322F20          STA     TPLEN ;STORE TAPE LENGTH
               ;  WE WISH THE ALARM TO BE SOUNDED 2
               ;  MINUTES BEFORE THE END OF THE TAPE.
               ;  THEREFORE SUBTRACT 2 FROM THE TAPE
               ;  LENGTH
110F D602            SUI     02D
               ;  CONVERT TO SECONDS
1111 4F              MOV  C, A      ;LOAD INTO REG C
1112 063C            MVI  B, 60D
1114 CD4508          CALL    MULT8 ;CONVERT TO SECONDS
               ;  BC NOW CONTAINS THE NUMBER OF
               ;  SECONDS. STORE IN THE CASSETTE COUNTER
1117 79              MOV  A, C
1118 322D20          STA     CASCT
111B 323120          STA     CSCT
111E 78              MOV  A, B
111F 322E20          STA     CASCT+1
1122 323220          STA     CSCT+1 ;STORE NEW VALUE
1125 C33901          JMP     KEYB  ;RETURN
               ;
               ;SUBROUTINE TO OUTPUT END OF FILE CHAR. TO TAPE
               ;  CLEAR CRDY. THE CASSETTE DRIVE IS NO
               ;  LONGER READY TO RECEIVE DATA
1128 AF        STP:  XRA  A
1129 322820          STA     CRDY  ;CLEAR READY FLAG
               ;  TURN ON THE CASSETTE DRIVE AND
               ;  ENABLE THE UART
112C 3E01            MVI  A, 01H
112E D30A            OUT     CPORT ;TURN ON CASSETTE DRIVE
1130 3E11            MVI  A, 11H
1132 D309            OUT     C8251 ;ENABLE UART
               ;  BUFER CONTAINS THE ADDRESS OF THE NEXT
               ;  AVAILABLE LOCATION IN THE CASSETTE BUFFER
1134 2A2520          LHLD    BUFER
               ;  LOAD FILE SEPARATORS
1137 CDBC11          CALL    LEOF  ;LOAD FILE SEPARATOR
               ;  SET THE END OF FILE CHARACTER TO BE THE
               ;  LAST CHARACTER TO BE SENT TO THE TAPE
113A 222A20          SHLD    OTEND ;LOAD END OF BUFFER
               ;  START THE DATA TRANSFER AT THE START
```

```
                  ;  OF THE BUFFER
113D 2A1D20             LHLD    CSTRT
1140 222320             SHLD    OUTD  ;LOAD CASS. DATA POINTER
1143 222520             SHLD    BUFER ;RESET BUFFER POINTER
                  ;  CFIN INDICATES WHETHER OR NOT THE CASSETTE
                  ;  TAPE IS FULL. CFIN IS SET SO THAT THE
                  ;  CASSETTE ROUTINE WILL RESET THE CASSETTE
                  ;  SYSTEM AFTER THE BUFFER IS LOADED ONTO TAPE
1146 3EFF               MVI  A, 0FFH
1148 322920             STA     CFIN  ;CASSETTE FINISHED
                  ;  DELAY 1 SECOND
                  ;  TO ALLOW THE DRIVE TO SPEED UP
114B CDC711             CALL    DEL1
                  ;  CASSETTE DRIVE IS NOW READY
                  ;  SET CRDY INDICATING THAT THE DRIVE IS
                  ;  READY TO RECEIVE DATA
114E 322820             STA     CRDY  ;ENABLE DATA TO CASSETTE
1151 C33901             JMP     KEYB  ;RETURN
                  ;
                  ;SUBROUTINE TO INITIALIZE CASS. TAPE INTERFACE
1154 2A3120       CIN:  LHLD    CSCT
1157 222D20             SHLD    CASCT ;INITIALIZE TAPE COUNTER
115A 2A1D20             LHLD    CSTRT
115D 222520             SHLD    BUFER ;INITIALIZE BUFFER POINTER
1160 C33901             JMP     KEYB  ;RETURN
                  ;
                  ;SUBROUTINE TO ENTER TEXT ONTO CASSETTE TAPE
                  ;CR (CTRL M) WILL CAUSE CARR RET AND LINE FEED
                  ;BS (CTRL H) WILL CAUSE A BACKSPACE
                  ;ETX (CTRL C) RETURN CONTROL TO THE MONITOR
1163 CD000A       CPR:  CALL    HOME  ;LINE FEED
1166 CD260A             CALL    RETN  ;CARRIAGE RETURN
1169 2A2520             LHLD    BUFER ;POINT HL AT BUFFER
116C CDB711             CALL    NLINE ;LOAD GROUP SEPARATOR
116F CDAE11             CALL    CRLF  ;CARR. RET. AND LINE FEED
1172 CDB711             CALL    NLINE ;LOAD GROUP SEPARATOR
                  ;  A CHARACTER IS RECEIVED FROM THE KEYBOARD
1175 CDC701       REC1: CALL    CHKEY ;READ KEYBOARD
1178 FE0D               CPI     CRET
117A CA8C11             JZ      CART1 ;CARRIAGE RETURN
                  ;  NOT A CARRIAGE RETURN
117D FE08               CPI     BAKSP
117F CA9B11             JZ      BACK1 ;BACKSPACE
                  ;  NOT A BACKSPACE
1182 FE03               CPI     ETX
1184 CAA511             JZ      RETRN
1187 77                 MOV  M, A     ;LOAD CHAR. INTO BUFFER
1188 23                 INX  H        ;INCREMENT BUFFER POINTER
1189 C37511             JMP     REC1  ;RECEIVE ANOTHER CHARACTER
                  ;  CARRIAGE RETURN HAS BEEN RECEIVED
118C CD000A       CART1: CALL   HOME  ;LINE FEED
118F CD260A             CALL    RETN  ;CARRIAGE RETURN
1192 CDAE11             CALL    CRLF  ;CARR. RET. AND LINE FEED
1195 CDB711             CALL    NLINE ;LOAD GROUP SEPARATOR
```

```
1198 C37511          JMP     REC1
               ;  BACKSPACE HAS BEEN RECEIVED
119B CDD009    BACK1: CALL    MOVL  ;BACKSPACE
119E CDD009           CALL    MOVL
11A1 2B               DCX  H
11A2 C37511           JMP     REC1
11A5 CDAE11    RETRN: CALL    CRLF  ;CARR. RET. & LINE FEED
11A8 222520           SHLD    BUFER ;RESTORE BUFFER POINTER
11AB C33901           JMP     KEYB  ;RETURN FROM ROUTINE
               ;
               ;SUBROUTINE TO PUT CARRIAGE RETURN ON TAPE
               ;LINE FEED IS SUPPLIED BY MTS
               ;  CARRIAGE RETURN IS PRECEEDED BY A DELETE.
               ;  HL IS ASSUMED TO CONTAIN THE BUFFER POINTER
11AE 3E7F      CRLF:  MVI  A, DELET
11B0 77               MOV  M, A     ;LOAD A DELETE
11B1 23               INX  H
11B2 3E0D             MVI  A, CRET
11B4 77               MOV  M, A     ;LOAD CARR. RETURN
11B5 23               INX  H
11B6 C9               RET
               ;
               ;SUBR. TO LOAD GROUP SEPARATOR FOR NEW LINE
               ;  HL IS ASSUMED TO CONTAIN THE BUFFER POINTER
11B7 3E1D      NLINE: MVI  A, 1DH
11B9 77               MOV  M, A     ;LOAD GROUP SEPARATOR
11BA 23               INX  H
11BB C9               RET
               ;
               ;SUBROUTINE TO LOAD FILE SEPARATORS
               ;  A DELETE PRECEEDS THE FILE SEPARATOR
11BC 3E7F      LEOF:  MVI  A, 7FH
11BE 77               MOV  M, A     ;LOAD DELETE
11BF 23               INX  H
11C0 3E1C             MVI  A, 1CH
11C2 77               MOV  M, A     ;LOAD FILE SEPARATOR
11C3 23               INX  H
11C4 77               MOV  M, A     ;LOAD FILE SEPARATOR
11C5 23               INX  H
11C6 C9               RET
               ;
               ;SUBROUTINE TO DELAY ONE SECOND
11C7 D5        DEL1:  PUSH D
11C8 C5               PUSH B
11C9 F5               PUSH PSW
11CA 0EFF             MVI  C, 0FFH
11CC 11FFFF    DEL2:  LXI  D, 0FFFFH ;LOAD COUNTER
11CF 1B        DEL3:  DCX  D          ;DECREMENT TIMER
11D0 7B               MOV  A, E
11D1 A7               ANA  A
11D2 C2CF11           JNZ     DEL3
11D5 7A               MOV  A, D
11D6 A7               ANA  A
11D7 C2CF11           JNZ     DEL3
```

```
                    ;  INSIDE LOOP IS DONE
11DA 79                  MOV  A, C
11DB A7                  ANA  A
11DC CAE411              JZ        ONSEC
11DF 0E00                MVI  C, 00H
11E1 C3CC11              JMP       DEL2  ;NOT DONE YET
11E4 F1             ONSEC: POP  PSW
11E5 C1                  POP  B
11E6 D1                  POP  D
11E7 C9                  RET
                    ;*********************************************
                    ;*                                           *
                    ;*      JUMP TABLE FOR KEYBOARD ENTRIES      *
                    ;*                                           *
                    ;*********************************************
11E8 414C4D         KYTAB: DB      'ALM'
11EB F210                DW        ALM
11ED 414E54              DB        'ANT'
11F0 F50A                DW        ANT
11F2 43454E              DB        'CEN'
11F5 DF10                DW        CEN
11F7 43484E              DB        'CHN'
11FA 720C                DW        CHN
11FC 43494E              DB        'CIN'
11FF 5411                DW        CIN
1201 434C52              DB        'CLR'
1204 740F                DW        CLR
1206 434F46              DB        'COF'
1209 0311                DW        COF
120B 434F4E              DB        'CON'
120E FC10                DW        CON
1210 435052              DB        'CPR'
1213 6311                DW        CPR
1215 435349              DB        'CSI'
1218 3D0F                DW        CSI
121A 435354              DB        'CST'
121D C910                DW        CST
121F 435450              DB        'CTP'
1222 0911                DW        CTP
1224 444953              DB        'DIS'
1227 5E0F                DW        DIS
1229 44504D              DB        'DPM'
122C 5802                DW        DPM
122E 445350              DB        'DSP'
1231 D603                DW        DSP
1233 464C47              DB        'FLG'
1236 320E                DW        FLG
1238 494E50              DB        'INP'
123B 2C03                DW        INP
123D 4D4C43              DB        'MLC'
1240 4C0F                DW        MLC
1242 4F5554              DB        'OUT'
1245 8303                DW        OUTP
1247 504354              DB        'PCT'
```

```
124A 670F          DW      PCT
124C 504F46        DB      'POF'
124F 2B0E          DW      POF
1251 50504D        DB      'PPM'
1254 8F0F          DW      PPM
1256 504F53        DB      'POS'
1259 3A0E          DW      POS
125B 505249        DB      'PRI'
125E 1A0E          DW      PRI
1260 505254        DB      'PRT'
1263 C80A          DW      PRT
1265 52554E        DB      'RUN'
1268 1503          DW      RUN
126A 534143        DB      'SAC'
126D C40B          DW      SAC
126F 534349        DB      'SCI'
1272 310F          DW      SCI
1274 53544D        DB      'STM'
1277 CD02          DW      STM
1279 535450        DB      'STP'
127C 2811          DW      STP
127E 54414E        DB      'TAN'
1281 550F          DW      TAN
1283 54494D        DB      'TIM'
1286 9F0E          DW      TIM
1288 545354        DB      'TST'
128B 410E          DW      TST
              ;*********************************************
              ;*                                           *
              ;*          INTERRUPT SERVICE SUBROUTINE     *
              ;*          FOR RST 6.5 (REAL TIME CLOCK)    *
              ;* THIS INTERRUPT ROUTINE ALSO CONTROLS THE  *
              ;* SCANNING OF THE ANTENNA SYSTEM.           *
              ;* BUFER  A POINTER TO THE NEXT AVAILABLE    *
              ;*        LOCATION IN THE BUFFER MEMORY      *
              ;* OUTD   A POINTER TO THE NEXT CHARACTER    *
              ;*        TO BE OUTPUT TO THE CASSETTE SYSTEM *
              ;* NDAT   THE NUMBER OF CHARACTERS CONTAINED *
              ;*        IN THE NEXT TWO READINGS           *
              ;*                                           *
              ;*********************************************
1800                ORG     1800H
1800 F5             PUSH    PSW
1801 C5             PUSH    B
1802 D5             PUSH    D
1803 E5             PUSH    H
              ;  SET PORT HIGH AT START OF ROUTINE
1804 3EFF           MVI  A, 0FFH
1806 D311           OUT     PORT
              ;  RESET FLIP FLOP & ENABLE INTERRUPTS
1808 3E40           MVI  A, 01000000B ;RESET FLIP FLOP
180A 30             DB      30H   ;SIM
180B 3EC0           MVI  A, 11000000B ;ENABLE FLIP FLOP
180D 30             DB      30H   ;SIM
```

```
180E FB              EI
                ;  INCREMENT REAL TIME CLOCK
180F 211720          LXI  H, SECO  ;POINT INDEX REG. AT CLOCK
1812 7E              MOV  A, M     ;READ SECO
1813 FE09            CPI     9
1815 CA1C18          JZ      GSEC1 ;SECO IS 9
1818 34              INR     M     ;INCREMENT SECO
1819 C36C18          JMP     DONE
                ;  TIME IS XX:XX:X9
                ;  CLEAR LS DIGIT OF SECOND
                ;  INCREMENT MS DIGIT OF SECONDS
181C AF         GSEC1: XRA   A
181D 77              MOV  M, A     ;CLEAR SECO
181E 23              INX     H
181F 7E              MOV  A, M     ;READ SEC1
1820 FE05            CPI     5
1822 CA2918          JZ      GMINO ;SEC1 IS 5
1825 34              INR     M     ;INCREMENT SEC1
1826 C36C18          JMP     DONE
                ;  TIME IS XX:XX:59
                ;  CLEAR MS DIGIT OF SECONDS
                ;  INCREMENT LS DIGIT OF MINUTES
1829 AF         GMINO: XRA   A
182A 77              MOV  M, A     ;CLEAR SEC1
182B 23              INX     H
182C 7E              MOV  A, M     ;READ MINO
182D FE09            CPI     9
182F CA3618          JZ      GMIN1 ;MINO IS 9
1832 34              INR     M     ;INCREMENT MINO
1833 C36C18          JMP     DONE
                ;  TIME IS XX:X9:59
                ;  CLEAR LS DIGIT OF MINUTES
                ;  INCREMENT MS DIGIT OF MINUTES
1836 AF         GMIN1: XRA   A
1837 77              MOV  M, A     ;CLEAR MINO
1838 23              INX     H
1839 7E              MOV  A, M     ;READ MIN1
183A FE05            CPI     5
183C CA4318          JZ      GHRO  ;MIN1 IS 5
183F 34              INR     M     ;INCREMENT MIN1
1840 C36C18          JMP     DONE
                ;  TIME IS XX:59:59
                ;  CLEAR MS DIGIT OF MINUTES
                ;  INCREMENT LS DIGIT OF HOUR
1843 AF         GHRO:  XRA   A
1844 77              MOV  M, A     ;CLEAR MIN1
1845 23              INX     H
1846 7E              MOV  A, M     ;READ HRO
1847 FE03            CPI     3
1849 CA5C18          JZ      GTHRE ;HRO IS 3
184C FE09            CPI     9
184E CA5518          JZ      GHR1  ;HRO IS 9
1851 34              INR     M     ;INCREMENT HRO
1852 C36C18          JMP     DONE
```

```
                  ;  TIME IS X9:59:59
                  ;  CLEAR LS DIGIT OF HOUR
                  ;  INCREMENT MS DIGIT OF HOUR
1855 AF           GHR1:  XRA     A
1856 77                  MOV  M, A     ;CLEAR HRO
1857 23                  INX     H
1858 34                  INR     M     ;INCREMENT HR1
1859 C36C18              JMP     DONE
                  ;  TIME IS X3:59:59
185C 23           GTHRE: INX     H
185D 7E                  MOV  A, M     ;READ HR1
185E FE02                CPI     2
1860 CA6818              JZ      ZERO  ;HR1 IS 2
1863 2B                  DCX     H
1864 34                  INR     M     ;INCREMENT HRO
1865 C36C18              JMP     DONE
                  ;  RESET CLOCK TO ZERO
1868 AF           ZERO:  XRA     A
1869 77                  MOV  M, A     ;CLEAR HR1
186A 2B                  DCX     H
186B 77                  MOV  M, A     ;CLEAR HRO
                  ;  REAL TIME CLOCK HAS BEEN INCREMENTED
                  ;  CHECK FOR ALARM CONDITION
186C 3A3020       DONE:  LDA     ALRM  ;CHECK ALARM FLAG
186F A7                  ANA  A        ;SET FLAGS
1870 C27D18              JNZ     CALRM ;ALARM IS ON
                  ;  ALARM IS NOT ON,
                  ;  CHECK IF ALARM HAS BEEN ACKNOWLEDGED
1873 3A2920              LDA     CFIN  ;HAS ALARM BEEN ACK.
1876 A7                  ANA  A        ;SET FLAGS
1877 C2BA18              JNZ     DON1  ;ALARM HAS BEEN ACK.
                  ;  NO ALARM CONDITION EXISTS
187A C39A18              JMP     DON2
                  ;  ALARM IS ON, SEE IF DISABLE SWITCH IS SET
187D 1           +CALRM: READ    ALARM ;CHECK ALARM SWITCH
187D 1 D5        +       PUSH    D     ;SAVE DE REGISTERS
187E 1 1603      +       MVI  D, ALARM  ;STORE SWITCH NUMBER IN D
1880 1 DB06      +       IN      SWCH  ;READ THE SWITCH INPUTS
1882 1 1F        +NO:    RAR           ;ROTATE DO INTO CARRY
1883 1 15        +       DCR  D        ;DECREMENT COUNTER
1884 1 C28218    +       JNZ     NO    ;NOT CORRECT SWITCH YET
1887 1 D1        +       POP     D     ;RESTORE DE REGISTERS
1888 DABA18              JC      DON1  ;ALARM STILL ON
                  ;  ALARM HAS BEEN DISABLED
                  ;  SET END OF FILE FLAG. END OF FILE
                  ;  CHARACTERS WILL BE LOADED ONTO TAPE.
188B 3EFF                MVI  A, OFFH
188D 323320              STA     EFILE
                  ;  ALARM HAS BEEN ACKNOWLEDGED
1890 322920              STA     CFIN
                  ;  ALARM IS NO LONGER ON
1893 AF                  XRA  A
1894 323020              STA     ALRM  ;CLEAR ALARM FLAG
1897 C3BA18              JMP     DON1
```

```
                 ;  NO ALARM CONDITION EXISTS
                 ;  IS CASSETTE SYSTEM ON?
189A 3A2C20      DON2:  LDA     CASON ;IS CASSETTE ON?
189D A7                 ANA  A        ;SET FLAGS
189E CABA18             JZ      DON1  ;CASSETTE NOT ON
                 ;  CASSETTE SYSTEM IS ON, DECREMENT TIMER
18A1 2A2D20             LHLD    CASCT ;LOAD CASSETTE TIMER
18A4 2B                 DCX  H        ;DECREMENT TIMER
18A5 222D20             SHLD    CASCT ;STORE TIMER
                 ;  CHECK TIMER
18A8 7C                 MOV  A, H
18A9 A7                 ANA  A        ;SET FLAGS
18AA F2BA18             JP      DON1  ;TAPE NOT FULL
                 ;  TAPE IS FULL, SOUND ALARM
18AD 3E03               MVI  A, 03H
18AF D30A               OUT     CPORT
18B1 3E01               MVI  A, 01H
18B3 D30A               OUT     CPORT ;TOGGLE ALARM FLIP FLOP
                 ;  ALARM IS ON
18B5 3EFF               MVI  A, 0FFH
18B7 323020             STA     ALRM  ;SET ALARM FLAG
                 ;  IS SCANNING FLAG ACTIVE?
18BA 3A1620      DON1:  LDA     SCAN  ;LOAD SCAN FLAG
18BD A7                 ANA  A        ;SET FLAGS
18BE CA171A             JZ      EXIT  ;DO NOT SCAN
                 ;  SCANNING FLAG IS ACTIVE
                 ;  CHECK IF SCAN INTERVAL IS OVER
18C1 3A3422             LDA     SCAN2
18C4 3D                 DCR  A
18C5 323422             STA     SCAN2 ;DECREMENT COUNTER
18C8 C2171A             JNZ     EXIT  ;DO NOT SCAN YET
                 ;  SCAN THIS TIME, RESET SCAN COUNTER
18CB 3A3322             LDA     SCAN1
18CE 323422             STA     SCAN2 ;RESET COUNTER
                 ;  CHNL CONTAINS THE CHANNEL NUMBER SET
                 ;  BY THE CHN INSTRUCTION. WHEN THE SYSTEM
                 ;  IS SET TO SCAN, THE CHANNEL WILL BE
                 ;  CHANGED. THE CHANNEL NUMBER IS
                 ;  THEREFORE CLEARED SO THE DISPLAY
                 ;  ROUTINE CAN INDICATE THAT NO CHANNEL
                 ;  IS SET.
18D1 AF                 XRA  A
18D2 323520             STA     CHNL  ;CHANNEL WILL BE CHANGED
                 ;  SCAN ANTENNAS AND STORE DATA
18D5 213B20             LXI  H, TABLE ;POINT HL AT TABLE
                 ;  SET ANTENNA NUMBER TO 0, SET X/Y BIT
18D8 3E80               MVI  A, 80H
18DA 321321             STA     ANTNO ;INITIALIZE ANTENNA CODE
18DD 0E05        STRT:  MVI  C, 05    ;LOAD CHANNEL COUNTER
18DF 113620             LXI  D, CHN1  ;POINT DE AT CHANNEL TABLE
                 ;  CHECK TO SEE IF THERE ARE ANY
                 ;  ACTIVE CHANNELS
18E2 1A                 LDAX D        ;LOAD CHANNEL CODE
18E3 A7                 ANA  A        ;SET FLAGS
```

```
18E4 CA3F19          JZ      NCH   ;NO ACTIVE CHANNELS
                 ;  TURN ON THE CHANNEL
18E7 3A1321          LDA     ANTNO
18EA D310            OUT     ANTA  ;TURN ON RELAY
                 ;  DELAY 2 MSEC
18EC 3EB0            MVI  A, 0B0H
18EE D300            OUT     MODEA ;SET TIMER MODE
18F0 3EF4            MVI  A, 0F4H
18F2 D301            OUT     TIMA2 ;OUTPUT LS BYTE
18F4 3E01            MVI  A, 01H
18F6 D301            OUT     TIMA2 ;OUTPUT MS BYTE
18F8 1        +TSTT1: READ    TTST
18F8 1 D5     +       PUSH    D     ;SAVE DE REGISTERS
18F9 1 1607   +       MVI  D, TTST  ;STORE SWITCH NUMBER IN D
18FB 1 DB06   +       IN      SWCH  ;READ THE SWITCH INPUTS
18FD 1 1F     +NO:    RAR           ;ROTATE D0 INTO CARRY
18FE 1 15     +       DCR  D        ;DECREMENT COUNTER
18FF 1 C2FD18 +       JNZ     NO    ;NOT CORRECT SWITCH YET
1902 1 D1     +       POP     D     ;RESTORE DE REGISTERS
1903 D2F818           JNC     TSTT1 ;TIMER NOT DONE YET
                  ;  DELAY IS OVER, ANTENNA IS READY
                  ;  OUTPUT THE CHANNEL TO THE RADIO
1906 1A           ANTON: LDAX D     ;LOAD CHANNEL CODE
1907 A7                ANA  A       ;SET FLAGS
1908 CA3F19            JZ      NCH  ;NO ACTIVE CHANNEL
190B D314              OUT     CHANL ;OUTPUT CHANNEL CODE
                  ;  TURN OF THE FILTER, THE FILTER WILL
                  ;  AUTOMATICALLY TURN BACK ON BEFORE
                  ;  THE DELAY IS OVER.
190D D315              OUT     FILTR ;TURN OFF FILTER
                  ;  DELAY 3 MSEC
190F 3EB0              MVI  A, 0B0H
1911 D300              OUT     MODEA ;SET TIMER MODE
1913 3EEE              MVI  A, 0EEH
1915 D301              OUT     TIMA2 ;OUTPUT LS BYTE
1917 3E02              MVI  A, 02H
1919 D301              OUT     TIMA2 ;OUTPUT MS BYTE
191B 1        +TSTT:   READ    TTST
191B 1 D5     +        PUSH    D     ;SAVE DE REGISTERS
191C 1 1607   +        MVI  D, TTST  ;STORE SWITCH NUMBER IN D
191E 1 DB06   +        IN      SWCH  ;READ THE SWITCH INPUTS
1920 1 1F     +NO:     RAR           ;ROTATE D0 INTO CARRY
1921 1 15     +        DCR  D        ;DECREMENT COUNTER
1922 1 C22019 +        JNZ     NO    ;NOT CORRECT SWITCH YET
1925 1 D1     +        POP     D     ;RESTORE DE REGISTERS
1926 D21B19            JNC     TSTT  ;TIMER NOT DONE YET
                   ;  DELAY IS OVER
                   ;  START A/D CONVERTER
1929 D312               OUT     STCON ;START CONVERSION
                   ;  WAIT FOR A/D CONVERTER TO FINISH
192B 1        +RADC:    READ    EOC   ;CONVERSION DONE?
192B 1 D5     +         PUSH    D     ;SAVE DE REGISTERS
192C 1 1604   +         MVI  D, EOC   ;STORE SWITCH NUMBER IN D
192E 1 DB06   +         IN      SWCH  ;READ THE SWITCH INPUTS
```

```
1930 1 1F    +NO:     RAR               ;ROTATE D0 INTO CARRY
1931 1 15    +        DCR  D            ;DECREMENT COUNTER
1932 1 C23019 +       JNZ      NO       ;NOT CORRECT SWITCH YET
1935 1 D1    +        POP      D        ;RESTORE DE REGISTERS
1936 D22B19           JNC      RADC     ;CHECK AGAIN
              ;  CONVERSION IS FINISHED
1939 DB13             IN       ADCIN ;READ A/D CONVERTER
193B 2F               CMA
193C D680             SUI      80H   ;CONVERT TO TRUE BINARY
193E 77               MOV  M, A      ;STORE IN TABLE
              ;  POINT HL AT NEXT TABLE LOCATION
193F 23       NCH:    INX  H         ;INCREMENT POINTER
              ;  ARE THERE MORE CHANNELS TO CHECK
1940 0D               DCR  C         ;DECREMENT CHANNEL COUNTER
1941 CA4819           JZ       NCHA  ;THIS ANTENNA IS FINISHED
              ;  THERE ARE MORE CHANNELS TO CHECK
1944 13               INX  D         ;INCREMENT CHANNEL POINTER
1945 C30619           JMP      ANTON ;READ NEXT CHANNEL
              ;  ALL CHANNELS HAVE BEEN CHECKED, ARE
              ;  ALL ANTENNAS DONE?
1948 3A1321   NCHA:   LDA      ANTNO ;READ ANTENNA NUMBER
194B 3C               INR  A
194C 321321           STA      ANTNO ;INCREMENT ANTENNA NUMBER
              ;  ARE X ANTENNAS DONE?
194F FE95             CPI  95H       ;ANTENNA X21?
1951 CA5C19           JZ       DOY   ;X ANTENNAS ARE DONE
              ;  ARE Y ANTENNAS DONE?
1954 FE16             CPI      22    ;ANTENNA Y22 (TEST)?
1956 CA6319           JZ       LDAT  ;ALL ANTENNAS ARE DONE
              ;  NOT FINISHED, CHECK NEXT ANTENNA
1959 C3DD18           JMP      STRT
              ;  SET ANTENNA NUMBER TO 0,
              ;  CLEAR THE X/Y BIT.
195C AF       DOY:    XRA  A
195D 321321           STA      ANTNO ;POINT AT Y RELAYS
1960 C3DD18           JMP      STRT
              ;  SCANNING IS FINISHED
              ;  CALCULATE TRANSMITTER POSITIONS
              ;  LOAD TRANSMITTER POSITIONS INTO
              ;  TEMPORARY BUFFER
1963 CD601B   LDAT:   CALL     LOADA ;LOAD LOCATIONS
              ;  LOAD TIME INTO TEMPORARY BUFFER
1966 CDBE1B           CALL     LDTME ;LOAD TIME
              ;  SET ACTIVITY FLAGS
1969 CD2F1C           CALL     ACTAD ;SET ACTIVITY
              ;  IF THERE IS A TRANSMITTER WITHIN DISTC
              ;  OF THE MASTER TRANSMITTER, INCREASE
              ;  THE CASSETTE STORE RATE.
196C CD031B           CALL     CSINC ;SHOULD RATE BE INCREASED
              ;  SHOULD THE CASSETTE DATA BE LOADED
              ;  DURING THIS SCAN INTERVAL
196F 3A3722           LDA      CASI2
1972 3D               DCR  A
1973 323722           STA      CASI2 ;DECREMENT CASS. COUNTER
```

```
1976 C2A519          JNZ      ISON  ;DO NOT STORE DATA
                ;  CASSETTE INTERVAL OVER, RESET COUNTER
1979 3A3622          LDA      CASI1
197C 323722          STA      CASI2 ;RESET CASSETTE COUNTER
                ;  SET FLAGS
197F CD151F          CALL     STTAN ;TEST ANTENNA FLAGS
1982 CD411F          CALL     STABL ;ABOVE/BELOW FLAGS
                ;  LOAD CASSETTE BUFFER WITH ANY
                ;  DATA WHICH HAS CHANGED
1985 CD1F1A          CALL     LODEC ;LOAD DATA ONTO CASSETTE
                ;  SHOULD THE DATA BE PRINTED
1988 3A2F22          LDA      PROF
198B A7              ANA  A         ;SET FLAGS
198C CAA519          JZ       ISON  ;DO NOT PRINT
                ;  PRINTER IS ENABLED, SHOULD THE DATA
                ;  BE PRINTED DURING THIS SCAN INTERVAL
198F 3A3222          LDA      PINT1
1992 3D              DCR  A
1993 323222          STA      PINT1  ;DEC. PRINTER COUNTER
1996 C2A519          JNZ      ISON  ;DO NOT PRINT
                ;  PRINT INTERVAL IS OVER, RESET COUNTER
1999 3A3122          LDA      PINT
199C 323222          STA      PINT1 ;RESET PRINTER COUNTER
                ;  MOVE TEMPORARY BUFFER INTO THE
                ;  PRINT BUFFER
199F CD5E1E          CALL     LDPRT ;LOAD PRINTER BUFFER
                ;  PRINT THE DATA
19A2 CD641F          CALL     PTINT ;PRINT DATA
                ;  IS CASSETTE READY TO RECEIVE DATA?
19A5 3A2820     ISON: LDA     CRDY
19A8 A7              ANA  A         ;SET FLAGS
19A9 CADE19          JZ       NRDY  ;DRIVE ON BUT NOT READY
                ;  CHECK FOR FULL BUFER
19AC 2A2520          LHLD     BUFER
19AF EB              XCHG
19B0 2A2120          LHLD     NDAT
19B3 19              DAD  D         ;ADD NDAT & BUFER
19B4 3A1F20          LDA      CEND  ;LOAD CEND(LOW)
19B7 2F              CMA
19B8 5F              MOV  E, A      ;LOAD E
19B9 3A2020          LDA      CEND+1 ;LOAD CEND(HIGH)
19BC 2F              CMA
19BD 57              MOV  D, A      ;LOAD D
19BE 13              INX  D         ;NEGATE CEND
19BF 19              DAD  D         ;ADD TO HL
19C0 7C              MOV  A, H
19C1 07              RLC            ;LOAD CARRY BIT
19C2 DA171A          JC       EXIT  ;BUFER+NDAT<CEND
                ;  BUFFER IS FULL, INITIALIZE CASSETTE SYSTEM
19C5 3E01            MVI  A, 01D
19C7 322720          STA      CCNT  ;LOAD COUNTER
19CA 3E01            MVI  A, 01H
19CC D30A            OUT      CPORT ;TURN ON CASSETTE
19CE AF              XRA  A
```

```
19CF 322820          STA     CRDY  ;DRIVE ON BUT NOT READY
19D2 3EFF            MVI  A, 0FFH
19D4 322C20          STA     CASON ;DRIVE IS ON
             ;  INITIALIZE CASSETTE INTERFACE
19D7 3E11            MVI  A, 11H
19D9 D309            OUT     C8251 ;SET COMMAND WORD
19DB C3171A          JMP     EXIT  ;EXIT
             ;  CCNT IS DECREMENTED UNTIL IT REACHES
             ;  ZERO. WHEN IT REACHES ZERO, THE
             ;  CASSETTE BUFFER IS INITIALIZED.
             ;  AFTER THIS, THE DATA IS BEING OUTPUT
             ;  TO THE CASSETTE DRIVE, AND THE ROUTINE
             ;  WILL EXIT HERE SINCE THE COUNTER
             ;  WILL THEN BE LESS THAN ZERO.
19DE 3A2720  NRDY:   LDA     CCNT
19E1 3D              DCR  A
19E2 322720          STA     CCNT  ;WAIT 1 SECOND
19E5 C2171A          JNZ     EXIT  ;1 SECOND NOT UP
             ;  CASSETTE IS NOW READY TO RECEIVE DATA
19E8 2A2520          LHLD    BUFER
             ;  CHECK FOR END OF FILE
             ;  END OF FILE FLAG IS SET IF THE
             ;  ALARM HAS BEEN RESET.
19EB 3A3320          LDA     EFILE
19EE A7              ANA  A        ;SET FLAGS
19EF CAFC19          JZ      LRS   ;NOT END OF FILE
             ;  END OF FILE, LOAD FILE SEPARATORS
19F2 CD7F1F          CALL    LEOF
19F5 AF              XRA  A
19F6 323320          STA     EFILE ;CLEAR EFILE
19F9 C3061A          JMP     INITC
             ;  LOAD RECORD SEPARATORS
19FC 3E7F    LRS:    MVI  A, 7FH
19FE 77              MOV  M, A     ;LOAD DELETE
19FF 23              INX  H
1A00 3E1E            MVI  A, 1EH
1A02 77              MOV  M, A     ;LOAD RECORD SEPARATOR
1A03 23              INX  H
1A04 77              MOV  M, A     ;LOAD RECORD SEPARATOR
1A05 23              INX  H
             ;  CASSETTE BUFFER IS READY
             ;  SET POINTERS FOR CASSETTE INTERRUPT
             ;  ROUTINE.
             ;  OTEND POINTS TO THE LAST DATA IN BUFFER
             ;  OUTD POINTS TO THE FIRST DATA IN BUFFER
1A06 222A20  INITC:  SHLD    OTEND ;LOAD OTEND
1A09 2A1D20          LHLD    CSTRT
1A0C 222320          SHLD    OUTD  ;LOAD OUTD
1A0F 222520          SHLD    BUFER ;RESET BUFER
1A12 3EFF            MVI  A, 0FFH
1A14 322820          STA     CRDY  ;DRIVE IS READY
             ;  RESET PORT AT END OF ROUTINE
1A17 AF      EXIT:   XRA  A
1A18 D311            OUT     PORT
```

```
1A1A E1               POP     H
1A1B D1               POP     D
1A1C C1               POP     B
1A1D F1               POP     PSW
1A1E C9               RET
             ;*********************************************
             ;*                                           *
             ;*            SCAN SUBROUTINES               *
             ;*                                           *
             ;*********************************************
             ;
             ;SUBROUTINE TO COMPARE LOCATIONS AND
             ;  ACTIVITIES WITH THE PREV. VALUES, AND LOAD
             ;  THE CASSETTE BUFFER WITH ANY VALUES WHICH
             ;  HAVE CHANGED.
1A1F F5      LODEC: PUSH PSW
1A20 C5             PUSH B
1A21 3A1B21         LDA     CNUM
1A24 47             MOV  B, A       ;STORE NUMBER OF CHANNELS
1A25 3E01           MVI  A, 1
1A27 CD321A  LDC1:  CALL    LOADC
1A2A 3C             INR  A          ;INCREMENT COUNTER
1A2B 05             DCR  B
1A2C C2271A         JNZ     LDC1    ;NOT DONE YET
1A2F C1             POP  B
1A30 F1             POP  PSW
1A31 C9             RET
             ;
             ;SUBROUTINE TO COMPARE LOCATIONS AND ACTIVITY
             ;  STATES WITH THE PREVIOUS VALUES.
             ;  IF EITHER HAS CHANGED, OUTPUT DATA LINE TO
             ;  THE CASSETTE BUFFER.
             ;  ENTER WITH THE CHANNEL NUMBER IN REGISTER A.
1A32 E5      LOADC: PUSH H
1A33 D5             PUSH D
1A34 C5             PUSH B
1A35 F5             PUSH PSW
             ;  I1T? WILL BE INCEMENTED BY THE VALUE
             ;       IN THE ACCUMULATOR
             ;  I22T? WILL BE INCREMENTED BY 22 TIMES
             ;       THE NUMBER IN THE ACCUMULATOR
             ;
1A36 212821         LXI  H, ACCNT
1A39 223321         SHLD    I1T3   ;LOAD ACTIVITY COUNTER
1A3C 216C21         LXI  H, LCX1
1A3F 223921         SHLD    I22T1  ;LOAD CURRENT LOCATION
1A42 217421         LXI  H, FLG1
1A45 223B21         SHLD    I22T2  ;LOAD CURRENT ACTIVITY
1A48 215D21         LXI  H, PRFL1
1A4B 223121         SHLD    I1T2   ;LOAD PREVIOUS ACTIVITY
1A4E CDDE1E         CALL    INPNT
1A51 F5             PUSH PSW
             ;  POINT DE AT THE PREVIOUS VALUES
1A52 113F21         LXI  D, PRX1
1A55 3D      LODC9: DCR  A
```

```
1A56 CA601A          JZ      LDC9
1A59 13              INX  D
1A5A CD561F          CALL    INCD5 ;INCREMENT DE 6 TIMES
1A5D C3551A          JMP     LODC9
1A60 F1       LDC9:  POP  PSW        ;RESTORE COUNTER
1A61 EB              XCHG
1A62 222F21          SHLD    I1T1  ;PREVIOUS LOCATION
              ;  POINTERS NOW POINT TO CORRECT LOCATIONS
              ;  I1T1 POINTS AT THE PREVIOUS LOCATIONS
              ;  I22T1 POINTS AT THE CURRENT LOCATIONS
              ;  I1T2 POINTS AT THE PREVIOUS ACTIVITY
              ;  I22T2 POINTS AT THE CURRENT ACTIVITY
              ;  I1T3 POINTS AT THE ACTIVITY COUNTER
              ;  COMPARE PRESENT LOCATION WITH THE
              ;  PREVIOUS LOCATION
              ;  B COUNTS THE NUMBER OF VALUES (X AND Y)
              ;  C COUNTS THE NUMBER OF DIGITS IN EACH
              ;  VALUE
1A65 0602            MVI  B, 2     ;LOAD COUNTER
1A67 0E03     LODC1: MVI  C, 3     ;LOAD COUNTER
1A69 2A3921   LODC2: LHLD    I22T1
1A6C 7E              MOV  A, M     ;LOAD CURRENT LOCATION
1A6D 23              INX  H
1A6E 223921          SHLD    I22T1 ;POINT AT NEXT DIGIT
1A71 2A2F21          LHLD    I1T1  ;POINT AT PREV. LOCATION
1A74 BE              CMP  M        ;CURRENT - PREVIOUS
1A75 C2991A          JNZ     NTSM1 ;NOT THE SAME
              ;  THIS DIGIT IS THE SAME
1A78 23              INX  H
1A79 222F21          SHLD    I1T1  ;INCREMENT POINTER
1A7C 0D              DCR  C        ;DECREMENT COUNTER
1A7D C2691A          JNZ     LODC2 ;COMPARE NEXT DIGIT
              ;  ALL DIGITS HAVE BEEN CHECKED.
1A80 2A3921          LHLD    I22T1
1A83 23              INX  H        ;POINT AT Y DATA
1A84 223921          SHLD    I22T1
1A87 05              DCR  B
1A88 C2671A          JNZ     LODC1 ;CHECK Y DATA
              ;  THE LOCATION HAS NOT CHANGED
              ;  HAS THE ACTIVITY CHANGED?
1A8B 2A3B21          LHLD    I22T2
1A8E 7E              MOV  A, M     ;LOAD CURRENT ACTIVITY
1A8F 2A3121          LHLD    I1T2  ;POINT AT PREV. ACTIVITY
1A92 BE              CMP  M        ;CURRENT - PREVIOUS
1A93 C2A31A          JNZ     NTSM2 ;NOT THE SAME
              ;  NEITHER THE LOCATION NOR THE
              ;  ACTIVITY HAVE CHANGED.
1A96 C3A81A          JMP     STRPR ;STORE DATA
              ;  THE LOCATION HAS CHANGED
1A99 2A3B21   NTSM1: LHLD    I22T2
1A9C 3631            MVI  M, 31H   ;SET ACTIVITY TO ONE
1A9E 2A3321          LHLD    I1T3
1AA1 3605            MVI  M, 5     ;RESET ACTIVITY COUNTER
              ;  LOCATION OR ACTIVITY HAS CHANGED
```

```
                   ;  LDCAS WILL STORE THE DATA IN THE
                   ;  CASSETTE BUFFER. TRANSMITTER NUMBER
                   ;  IS ENTERED IN REGISTER A.
1AA3 F1            NTSM2: POP  PSW
1AA4 F5                   PUSH PSW        ;RESTORE CHANNEL NUMBER
1AA5 CDB11E               CALL     LDCAS ;LOAD CASSETTE BUFFER
                   ;  MOVE CURRENT DATA TO PREVIOUS FILE
1AA8 F1            STRPR: POP  PSW
1AA9 F5                   PUSH PSW        ;RESTORE CHANNEL NUMBER
1AAA 216C21               LXI  H, LCX1
1AAD 223921               SHLD     I22T1 ;LOAD CURRENT LOCATION
1AB0 217421               LXI  H, FLG1
1AB3 223B21               SHLD     I22T2 ;LOAD CURRENT ACTIVITY
1AB6 215D21               LXI  H, PRFL1
1AB9 223121               SHLD     I1T2  ;LOAD PREVIOUS ACTIVITY
1ABC CDDE1E               CALL     INPNT
1ABF F5                   PUSH PSW
1AC0 113F21               LXI  D, PRX1
1AC3 3D            LODC8: DCR  A
1AC4 CACE1A               JZ       LDC8
1AC7 13                   INX  D
1AC8 CD561F               CALL     INCD5 ;INCREMENT DE 6 TIMES
1ACB C3C31A               JMP      LODC8
1ACE F1            LDC8:  POP  PSW         ;RESTORE COUNTER
1ACF EB                   XCHG
1AD0 222F21               SHLD     I1T1  ;PREVIOUS LOCATION
                   ;  POINTERS NOW POINT TO CORRECT LOCATIONS
                   ;  I1T1 POINTS AT THE PREVIOUS LOCATIONS
                   ;  I22T1 POINTS AT THE CURRENT LOCATIONS
                   ;  I1T2 POINTS AT THE PREVIOUS ACTIVITY
                   ;  I22T2 POINTS AT THE CURRENT ACTIVITY
                   ;  MOVE CURRENT LOCATION TO PREVIOUS FILE
1AD3 0602                 MVI  B, 2
1AD5 0E03          LODC4: MVI  C, 3
1AD7 2A3921        LODC5: LHLD     I22T1
1ADA 7E                   MOV  A, M       ;LOAD CURRENT LOCATION
1ADB 23                   INX  H
1ADC 223921               SHLD     I22T1 ;POINT AT NEXT DIGIT
1ADF 2A2F21               LHLD     I1T1  ;POINT AT PREV. LOCATION
1AE2 77                   MOV  M, A       ;STORE LOCATION
1AE3 23                   INX  H
1AE4 222F21               SHLD     I1T1  ;INCREMENT POINTER
1AE7 0D                   DCR  C          ;DECREMENT COUNTER
1AE8 C2D71A               JNZ      LODC5 ;NEXT DIGIT
1AEB 2A3921               LHLD     I22T1
1AEE 23                   INX  H
1AEF 223921               SHLD     I22T1 ;POINT AT Y DATA
1AF2 05                   DCR  B
1AF3 C2D51A               JNZ      LODC4 ;LOAD Y DATA
                   ;  MOVE ACTIVITY TO PREVIOUS FILE
1AF6 2A3B21               LHLD     I22T2
1AF9 7E                   MOV  A, M       ;LOAD CURRENT ACTIVITY
1AFA 2A3121               LHLD     I1T2  ;POINT AT PREV. ACTIVITY
1AFD 77                   MOV  M, A       ;STORE ACTIVITY
```

```
1AFE F1             POP  PSW
1AFF C1             POP  B
1B00 D1             POP  D
1B01 E1             POP  H
1B02 C9             RET
             ;SUBROUTINE TO CHECK IF THE CASSETTE RATE
             ;  SHOULD BE INCREASED. IF ANY CHANNELS
             ;  ARE WITHIN DISTC OF CHANNEL 1, THE
             ;  CASSETTE RATE WILL BE CHANGED TO THE
             ;  SCANNING RATE.  IF NO CHANNELS ARE
             ;  WITHIN RANGE THE CASSETTE RATE IS
             ;  RESET TO ITS PREVIOUS VALUE.  IF THERE
             ;  IS NO ACTIVITY ON THE CHANNEL WHICH
             ;  IS WITHIN RANGE, THE RATE IS NOT
             ;  INCREASED.
1B03 F5      CSINC: PUSH PSW
1B04 E5             PUSH H
1B05 C5             PUSH B
1B06 D5             PUSH D
             ;  IF DISTC IS SET TO ZERO, DO NOT
             ;  INCREASE THE CASSETTE STORE RATE.
1B07 3A1C21         LDA     DISTC
1B0A FE00           CPI     0
1B0C CA421B         JZ      CSIN3
             ;  RESET CASSETTE RATE TO ITS PREVIOUS
             ;  VALUE
1B0F 3A3522         LDA     CASIP
1B12 323622         STA     CASI1 ;RESET CASSETTE RATE
1B15 3A1B21         LDA     CNUM
1B18 47             MOV  B, A     ;LOAD B WITH NUM OF CHANN.
1B19 05             DCR  B
             ;  B CONTAINS THE NUMBER OF CHANNELS
             ;  TO BE CHECKED
1B1A 118A21         LXI  D, FLG2
1B1D 211C21         LXI  H, DISTC
1B20 3E02           MVI  A, 2     ;LOAD FIRST CHANNEL
             ;  DISTA WILL FIND THE DISTANCE BETWEEN
             ;  TRANSMITTER ONE AND THE TRANSMITTER
             ;  WHOSE NUMBER IS ENTERED IN REGISTER
             ;  A. HL POINTS TO THE SETPOINT.
1B22 F5      CSIN1: PUSH PSW
1B23 CD471B         CALL    DISTA ;FIND DISTANCE
1B26 BE             CMP  M        ;DISTANCE-SETPOINT
1B27 F2381B         JP      CSIN2 ;NOT WITHIN RANGE
             ;  ONE CHANNEL IS WITHIN RANGE
             ;  CHECK THE ACTIVITY
             ;  DE POINTS AT THE ACTIVITY FLAG
1B2A 1A             LDAX D
1B2B FE30           CPI     30H   ;NO ACTIVITY
1B2D CA381B         JZ      CSIN2 ;DO NOT CHANGE RATE
             ;  ACTIVITY IS A ONE, INCREASE THE RATE
1B30 3E01           MVI  A, 1     ;LOAD NEW SCAN INTERVAL
1B32 323622         STA     CASI1 ;LOAD CASSETTE RATE
1B35 323722         STA     CASI2
```

```
                 ;  CHECK THE NEXT TRANSMITTER
1B38 F1          CSIN2: POP  PSW        ;RELOAD COUNTER
1B39 3C                 INR  A          ;INCREMENT COUNTER
1B3A CD671F             CALL    INDE    ;INCREMENT DE REGISTER
1B3D 13                 INX  D          ;22 TIMES
                 ;  DE NOW POINTS AT ACTIVITY FLAG FOR
                 ;  THE NEXT TRANSMITTER.
1B3E 05                 DCR  B
1B3F C2221B             JNZ     CSIN1 ;NOT DONE YET
                 ;  ALL TRANSMITTERS HAVE BEEN CHECKED.
1B42 D1          CSIN3: POP  D
1B43 C1                 POP  B
1B44 E1                 POP  H
1B45 F1                 POP  PSW
1B46 C9                 RET
                 ;
                 ;SUBROUTINE TO CALC. THE DIST. BETWEEN CHAN. 1
                 ;  AND THE CHANNEL NUMBER ENTERED IN THE A REG.
                 ;  THE RESULT (IN ANTENNAS) IS RETURNED IN A
1B47 C5          DISTA: PUSH B
1B48 D5                 PUSH D
1B49 016C21             LXI  B, LCX1
1B4C 116C21             LXI  D, LCX1
1B4F 3D          DIST4: DCR  A          ;DECREMENT COUNTER
1B50 CA5A1B             JZ      DIST5
1B53 CD671F             CALL    INDE   ;INCREMENT DE 21 TIMES
1B56 13                 INX  D          ;22 TIMES
1B57 C34F1B             JMP     DIST4
                 ;  BC POINTS TO THE LOCATION OF TRANS. 1
                 ;  DE POINTS TO THE LOCATION OF TRANS. ?
                 ;  DISTC WILL CALCULATE THE DISTANCE
                 ;  AND RETURN THE ANSWER IN REGISTER A.
1B5A CDA51D      DIST5: CALL    DIST   ;CALCULATE DISTANCE
1B5D D1                 POP  D
1B5E C1                 POP  B
1B5F C9                 RET
                 ;
                 ;SUBROUTINE TO LOAD THE CHANNEL LOCATIONS
                 ;  INTO THE PRINT BUFFER
                 ;  THE POPULATION MATRIX IS INCREMENTED
1B60 F5          LOADA: PUSH PSW
1B61 D5                 PUSH D
1B62 C5                 PUSH B
                 ;  ACFL1 IS MOVED TO ACFL2
                 ;  ACFL1 CONTAINS THE PRESENT A/D READINGS
                 ;  ACFL2 CONTAINS THE PAST A/D READINGS.
1B63 3E05               MVI  A, 5      ;LOAD COUNTER
1B65 011E21             LXI  B, ACFL1
1B68 112321             LXI  D, ACFL2
1B6B F5          LDD1:  PUSH PSW        ;STORE COUNTER
1B6C 0A                 LDAX B
1B6D 12                 STAX D
1B6E 03                 INX  B
1B6F 13                 INX  D
```

```
1B70 F1              POP  PSW         ;RESTORE COUNTER
1B71 3D              DCR  A           ;DECREMENT COUNTER
1B72 C26B1B          JNZ      LDD1    ;NOT DONE YET
               ;  ACTIVITY READINGS HAVE BEEN MOVED
               ;  INTO THE PREVIOUS FILE
1B75 3A1B21          LDA      CNUM    ;LOAD NUMBER OF CHANNELS
1B78 47              MOV  B, A
1B79 3E01            MVI  A, 1        ;LOAD COUNTER
               ;  LDDT WILL LOAD ONE CHANNEL DATA INTO
               ;  THE PRINT BUFFER. TRANSMITTER NUMBER
               ;  IS ENTERED IN REGISTER A.
1B7B CDE71B    LOAD1: CALL    LDDT    ;LOAD CHANNEL DATA
               ;  FOR THE FIRST THREE TRANSMITTERS
               ;  INCREMENT THE POPULATION MATRIX.
1B7E FE04            CPI      4
1B80 F2861B          JP       LOAD2   ;DO NOT INCREMENT
1B83 CD8F1B          CALL     LDPOP   ;INCREMENT POP MATRIX
1B86 3C        LOAD2: INR  A          ;INCREMENT COUNTER
1B87 05              DCR  B           ;DECREMENT CHANNEL COUNTER
1B88 C27B1B          JNZ      LOAD1   ;LOAD NEXT CHANNEL DATA
               ;  ALL CHANNELS HAVE BEEN LOADED INTO
               ;  THE TEMPORARY BUFFER.
1B8B C1              POP  B
1B8C D1              POP  D
1B8D F1              POP  PSW
1B8E C9              RET
               ;
               ;SUBROUTINE TO INCREMENT POPULATION MATRIX
               ;  LOCATION TO BE INCREMENTED IS CONTAINED IN
               ;  NEARX AND NEARY.
               ;  ENTER WITH A CONTAINING CHANNEL NUMBER
1B8F D5        LDPOP: PUSH D
1B90 E5               PUSH H
1B91 C5               PUSH B
1B92 F5               PUSH PSW
1B93 21A025           LXI  H, POPM   ;POINT HL AT MATRIX
1B96 3D        LDPO1: DCR  A          ;DECREMENT CHANNEL NUMBER
1B97 CAA01B           JZ       LDPO2  ;DONE
1B9A CD821F           CALL     AD882  ;DPOINT HL AT NEXT CHANNEL
1B9D C3961B           JMP      LDPO1
               ;  HL NOW POINTS AT CORRECT CHANNEL TABLE
1BA0 3A3E22    LDPO2: LDA      NEARX
1BA3 47               MOV  B, A       ;STORE IN REGISTER B
1BA4 0E2A             MVI  C, 42
1BA6 CD701F           CALL     MULT8  ;NEARX*42
1BA9 09               DAD  B          ;ADD TO HL REGISTER
               ;  HL NOW POINTS AT X PORTION OF TABLE
1BAA 3A3D22           LDA      NEARY
1BAD 5F               MOV  E, A       ;LOAD DE WITH NEARY
1BAE 1600             MVI  D, 0
1BB0 19               DAD  D
1BB1 19               DAD  D          ;ADD 2*NEARY TO HL
               ;  HL NOW POINTS AT CORRECT LOCATION
1BB2 56               MOV  D, M
```

```
1BB3 23              INX  H
1BB4 5E              MOV  E, M
               ;  DE NOW CONTAINS THE COUNTER
1BB5 13              INX  D         ;INCREMENT THE COUNTER
1BB6 73              MOV  M, E
1BB7 2B              DCX  H
1BB8 72              MOV  M, D      ;RESTORE COUNTER
               ;  COUNTER HAS BEEN INCREMENTED.
1BB9 F1              POP  PSW
1BBA C1              POP  B
1BBB E1              POP  H
1BBC D1              POP  D
1BBD C9              RET
               ;
               ;SUBROUTINE TO LOAD TIME INTO THE PRINT BUFFER
1BBE F5        LDTME: PUSH PSW
1BBF D5               PUSH D
1BC0 C5               PUSH B
1BC1 E5               PUSH H
1BC2 0E05             MVI  C, 5      ;LOAD COUNTER
1BC4 216221           LXI  H, TM1    ;POINT AT FIRST TIME LOC.
1BC7 111C20    LDTM:  LXI  D, HR1    ;POINT DE AT TIMER
1BCA 0606             MVI  B, 6      ;LOAD DIGIT COUNTER
1BCC 1A        LDTM1: LDAX D         ;LOAD DIGIT
1BCD CD7C1F           CALL    CONAS  ;CONVERT TO ASCII
1BD0 77               MOV  M, A      ;STORE IN BUFFER
1BD1 1B               DCX  D         ;POINT TO NEXT DIGIT
1BD2 23               INX  H         ;POINT AT NEXT LOCATION
1BD3 05               DCR  B         ;DECREMENT DIGIT COUNTER
1BD4 C2CC1B           JNZ     LDTM1  ;STORE NEXT DIGIT
               ;  ONE SET OF DIGITS HAS BEEN LOADED
1BD7 CD5C1F           CALL    IHL20  ;INCREMENT HL 20 TIMES
1BDA 2B               DCX  H
1BDB 2B               DCX  H
1BDC 2B               DCX  H
1BDD 2B               DCX  H         ;POINT HL AT NEXT LOCATION
               ;  HL POINTS AT NEXT TIME LOCATION
               ;  IN THE TEMPORARY BUFFER
1BDE 0D               DCR  C         ;DECREMENT COUNTER
1BDF C2C71B           JNZ     LDTM   ;LOAD NEXT TIME
               ;  ALL FIVE DATA LINES HAVE THE
               ;  TIME ENTERED.
1BE2 E1               POP  H
1BE3 C1               POP  B
1BE4 D1               POP  D
1BE5 F1               POP  PSW
1BE6 C9               RET
               ;
               ;SUBROUTINE TO LOAD ONE CHANNEL LOCATION INTO
               ;  THE PRINT BUFFER.
               ;  A CONTAINS THE CHAN. TO BE LOADED (1 TO 5)
               ;  THE A/D VALUE FOR THE Y ANTENNAS IS
               ;  STORED IN THE ACTIVITY FILE
               ;  NEAREST ANTENNA IS STORED IN NEARX AND NEARY
```

```
1BE7 F5       LDDT:  PUSH PSW
1BE8 D5              PUSH D
1BE9 E5              PUSH H
1BEA C5              PUSH B
1BEB 011E21          LXI  B, ACFL1 ;POINT AT ACTIVITY FILE
1BEE 216C21          LXI  H, LCX1  ;POINT HL AT LOCATION
1BF1 113B20          LXI  D, TABLE
1BF4 3D       LDDT1: DCR  A        ;DECREMENT NUMBER
1BF5 CA021C          JZ      LDDT2 ;LOAD DATA
1BF8 13              INX  D        ;INCREMENT POINTER
1BF9 03              INX  B
1BFA CD5C1F          CALL    IHL20 ;INCREMENT HL
1BFD 23              INX  H
1BFE 23              INX  H        ;22 TIMES
1BFF C3F41B          JMP  LDDT1
              ;  DE NOW POINTS AT THE FIRST ANTENNA DATA
              ;  BC POINTS AT ACTIVITY FILE
              ;  HL POINTS AT LOCATION DATA
              ;  POSAX WILL FIND THE POSITION.
              ;  ANSWER IS RETURNED (IN ASCII) IN
              ;  LCY5 TO LCY5+2.
              ;  NEAREST ANTENNA IS RETURNED IN NEARY
              ;  A/D READING FOR THE MAXIMUM ANTENNA
              ;  IS RETURNED IN APRES.
1C02 C5       LDDT2: PUSH B
1C03 0602            MVI  B, 2
1C05 CDC71C          CALL    POSAX ;FIND POSITION
1C08 3AC821          LDA     LCY5
1C0B 77              MOV  M, A     ;STORE IN BUFFER
1C0C 23              INX  H
1C0D 3AC921          LDA     LCY5+1
1C10 77              MOV  M, A     ;LOAD SECOND DIGIT
1C11 23              INX  H
1C12 3ACA21          LDA     LCY5+2
1C15 77              MOV  M, A     ;LOAD THIRD DIGIT
1C16 23              INX  H
1C17 05              DCR  B        ;DECREMENT COUNTER
1C18 CA251C          JZ      LDDT3
              ;  X DATA HAS BEEN LOADED, LOAD Y DATA
1C1B 23              INX  H        ;POINT HL AT Y DATA
1C1C 3A3D22          LDA     NEARY
1C1F 323E22          STA     NEARX ;STORE X LOCATION
1C22 C3051C          JMP     LDDT2+3 ;LOAD DATA
              ;  X AND Y DATA HAVE BEEN STORED IN THE
              ;  TEMPORARY BUFFER. STORE A/D VALUE
              ;  IN THE ACTIVITY FILE.
1C25 C1       LDDT3: POP  B
1C26 3A3A22          LDA     APRES ;LOAD A/D VALUE
1C29 02              STAX B        ;STORE IN ACTIVITY FILE
1C2A C1              POP  B
1C2B E1              POP  H
1C2C D1              POP  D
1C2D F1              POP  PSW
1C2E C9              RET
```

```
                     ;
                     ;SUBROUTINE TO SET ALL ACTIVITY FLAGS
                     ;  BASED ON THE A/D VALUES ONLY
1C2F F5              ACTAD: PUSH PSW
1C30 3E01                   MVI  A, 1       ;LOAD COUNTER
                     ;  ACT WILL SET THE ACTIVITY FLAG FOR THE
                     ;  TRANSMITTER WHOSE NUMBER IS ENTERED
                     ;  IN THE A REGISTER.
1C32 CD3D1C          ACAD1: CALL     ACT    ;SET FLAGS
1C35 3C                     INR  A          ;INCREMENT COUNTER
1C36 FE06                   CPI  6
1C38 C2321C                 JNZ      ACAD1 ;NOT DONE YET
                     ;  ALL 5 CHANNELS HAVE BEEN CHECKED.
1C3B F1                     POP  PSW
1C3C C9                     RET
                     ;
                     ;SUBROUTINE TO SET ACTIVITY FLAGS BASED ON A/D
                     ;  VALUES ONLY. LATERAL MOTION SHOULD OVERRIDE
                     ;  THESE FLAGS AT A LATER TIME
                     ;  ENTER WITH THE CHANNEL NUMBER IN THE A REG.
                     ;  IF THERE IS NO MOTION FOR FIVE SCAN INT.
                     ;  THE ACTIVITY FLAG WILL BE SET TO ZERO.
1C3D C5              ACT:   PUSH B
1C3E D5                     PUSH D
1C3F E5                     PUSH H
1C40 F5                     PUSH PSW
1C41 212821                 LXI  H, ACCNT
1C44 222D21                 SHLD     ACPNT ;STORE COUNTER POINTER
1C47 117421                 LXI  D, FLG1   ;POINT AT FIRST ACT. FLAG
1C4A 211E21                 LXI  H, ACFL1
1C4D 012321                 LXI  B, ACFL2
1C50 3D              ACT1:  DCR  A         ;DECREMENT COUNTER
1C51 CA661C                 JZ       ACT2  ;DONE
1C54 CD671F                 CALL     INDE  ;INCREMENT DE 21 TIMES
1C57 13                     INX  D         ;22 TIMES
1C58 23                     INX  H
1C59 03                     INX  B
1C5A E5                     PUSH H
1C5B 2A2D21                 LHLD     ACPNT
1C5E 23                     INX  H
1C5F 222D21                 SHLD     ACPNT ;INCREMENT COUNTER POINTER
1C62 E1                     POP  H
1C63 C3501C                 JMP      ACT1
                     ;  HL NOW POINTS AT PRESENT ACTIVITY FILE
                     ;  BC NOW POINTS AT PAST ACTIVITY FILE
                     ;  DE POINTS AT THE ACTIVITY FLAG
                     ;  ACPNT POINTS AT THE ACTIVITY COUNTER
                     ;  ACPCT CONTAINS THE PERCENTAGE BY WHICH
                     ;  THE VALUE MAY CHANGE.
1C66 0A              ACT2:  LDAX B         ;LOAD PREVIOUS VALUE
1C67 BE                     CMP  M         ;PREVIOUS-PRESENT
1C68 FA8B1C                 JM       ACT3
                     ;  PREVIOUS>PRESENT
1C6B D5                     PUSH D
```

```
1C6C 47                 MOV  B, A      ;LOAD PREVIOUS VALUE
1C6D 3A1D21             LDA      ACPCT
1C70 4F                 MOV  C, A      ;LOAD PERCENTAGE
1C71 CD701F             CALL     MULT8 ;PREVIOUS*PERCENT
1C74 78                 MOV  A, B
1C75 2F                 CMA
1C76 57                 MOV  D, A
1C77 79                 MOV  A, C
1C78 2F                 CMA
1C79 5F                 MOV  E, A
1C7A 13                 INX  D         ;-(PREVIOUS*PERCENT)
1C7B 46                 MOV  B, M      ;LOAD PRESENT
1C7C 0E64               MVI  C, 100
1C7E CD701F             CALL     MULT8 ;PRESENT*100
1C81 EB                 XCHG
1C82 09                 DAD  B         ;PRES*100-(PREV*PERCENT)
1C83 7C                 MOV  A, H
1C84 A7                 ANA  A         ;SET FLAGS
1C85 F2A81C             JP       ACTZ  ;ACTIVITY IS ZERO
1C88 C3B81C             JMP      ACTO  ;ACTIVITY IS ONE
                ;  PRESENT>PREVIOUS
1C8B D5         ACT3:   PUSH D
1C8C 47                 MOV  B, A      ;LOAD PREVIOUS
1C8D 0E64               MVI  C, 100
1C8F CD701F             CALL     MULT8 ;PREVIOUS*100
1C92 78                 MOV  A, B
1C93 2F                 CMA
1C94 57                 MOV  D, A
1C95 79                 MOV  A, C
1C96 2F                 CMA
1C97 59                 MOV  E, C
1C98 13                 INX  D         ;-(PREVIOUS*100)
1C99 46                 MOV  B, M      ;LOAD PRESENT
1C9A 3A1D21             LDA      ACPCT
1C9D 4F                 MOV  C, A      ;LOAD PERCENTAGE
1C9E CD701F             CALL     MULT8 ;PRESENT*PERCENTAGE
1CA1 EB                 XCHG
1CA2 09                 DAD  B         ;PRES*PERCENT-(PREV*100)
1CA3 7C                 MOV  A, H
1CA4 A7                 ANA  A         ;SET FLAGS
1CA5 F2B81C             JP       ACTO  ;ACTIVITY IS A ONE
                ;  ACTIVITY IS ZERO
                ;  DECREMENT THE COUNTER
1CA8 D1         ACTZ:   POP  D
1CA9 2A2D21             LHLD     ACPNT ;POINT HL AT COUNTER
1CAC 7E                 MOV  A, M      ;LOAD COUNTER
1CAD 3D                 DCR  A
1CAE 77                 MOV  M, A      ;DECREMENT ACT. COUNTER
1CAF C2C21C             JNZ      ACT7  ;EXIT
                ;  ACTIVITY HAS BEEN ZERO FOR FIVE
                ;  SCAN INTERVALS, SET FLAG TO ZERO
1CB2 3E30               MVI  A, 30H
1CB4 12                 STAX D         ;CLEAR ACTIVITY FLAG
1CB5 C3C21C             JMP      ACT7  ;EXIT
```

```
                 ;  ACTIVITY IS A ONE,
                 ;  RESET COUNTER TO 5
1CB8 D1          ACTO:  POP   D
1CB9 3E31               MVI   A, 31H
1CBB 12                 STAX  D          ;SET ACTIVITY FLAG
1CBC 2A2D21             LHLD      ACPNT  ;POINT HL AT COUNTER
1CBF 3E05               MVI   A, 5
1CC1 77                 MOV   M, A       ;RESET ACTIVITY COUNTER
1CC2 F1          ACT7:  POP   PSW
1CC3 E1                 POP   H
1CC4 D1                 POP   D
1CC5 C1                 POP   B
1CC6 C9                 RET
                 ;
                 ;SUBROUTINE TO FIND POSITION ON ONE AXIS
                 ;  DE REGISTER POINTS TO FIRST ANTENNA DATA
                 ;  ANSWER IS RETURNED (IN ASCII) IN LCY5-LCY5+2
                 ;  NEAREST ANTENNA IS RETURNED IN NEARY
                 ;  A/D READING FOR MAX ANT IS RETURNED IN APRES
                 ;  MAXNO CONTAINS THE NUMBER OF THE
                 ;  ANTENNA WITH THE HIGHEST READING.
                 ;  ALAST CONTAINS THE A/D READING FOR
                 ;  THE ANTENNA IMMEDIATELY BEFORE THE
                 ;  ANTENNA WITH THE HIGHEST READING
                 ;  APRES CONTAINS THE A/D READING FOR
                 ;  THE ANTENNA WITH THE HIGHEST READING.
                 ;  STORE FLAG (REG B) IS SET IF A NEW
                 ;  MAXIMUM READING HAS BEEN FOUND.
1CC7 E5          POSAX: PUSH  H
1CC8 C5                 PUSH  B
1CC9 213A22             LXI   H, APRES
1CCC AF                 XRA   A
1CCD 323C22             STA       MAXNO  ;CLEAR MAX ANTENNA NUMBER
1CD0 323922             STA       ALAST  ;CLEAR PREVIOUS VALUE
1CD3 4F                 MOV   C, A       ;CLEAR ANTENNA NUMBER
1CD4 06FF               MVI   B, 0FFH    ;SET STORE FLAG
                 ;  READ FIRST ANTENNA, ASSUME THAT THIS
                 ;  IS THE LARGEST READING.
1CD6 1A                 LDAX  D
1CD7 323A22             STA       APRES  ;STORE MAX VALUE
1CDA 1A          POSA1: LDAX  D
1CDB 323822             STA       APREV  ;STORE VALUE
                 ;  READ NEXT ANTENNA
1CDE CD561F             CALL      INCD5  ;LOOK AT NEXT ANTENNA
1CE1 0C                 INR   C          ;INCREMENT ANTENNA NUMBER
1CE2 78                 MOV   A, B
1CE3 A7                 ANA   A          ;CHECK STORE FLAG
1CE4 1A                 LDAX  D          ;LOAD ANTENNA DATA
1CE5 CAEB1C             JZ        NSTOR  ;STORE FLAG NOT SET
                 ;  A NEW MAXIMUM WAS FOUND ON THE
                 ;  PREVIOUS ANTENNA, THEREFORE STORE
                 ;  THE DATA FOR THIS ANTENNA IN ANEXT
1CE8 323B22             STA       ANEXT
                 ;  CHECK THE PRESENT DATA TO SEE IF IT IS
```

```
                ;  A NEW MAXIMUM.
1CEB BE         NSTOR: CMP  M          ;PRESENT - PREVIOUS
1CEC FA011D            JM       NOMAX
                ;  A NEW MAXIMUM HAS BEEN FOUND
                ;  STORE THE VALUE IN APRES
1CEF 323A22            STA      APRES
                ;  SET THE FLAG SO THAT THE NEXT ANTENNA
                ;  VALUE WILL BE STORED IN ANEXT
1CF2 06FF              MVI  B, OFFH  ;SET STORE FLAG
                ;  STORE THE PREVIOUS VALUE IN ALAST
1CF4 3A3822            LDA      APREV
1CF7 323922            STA      ALAST ;STORE LAST VALUE
                ;  STORE THE ANTENNA NUMBER
1CFA 79                MOV  A, C      ;STORE ANTENNA NUMBER
1CFB 323C22            STA      MAXNO ;STORE ANTENNA NUMBER
1CFE C3031D            JMP      POSA2
                ;  THE CURRENT VALUE IS NOT A NEW
                ;  MAXIMUM
1D01 0600       NOMAX: MVI  B, 00H    ;CLEAR STORE FLAG
                ;  HAVE ALL THE ANTENNAS BEEN CHECKED.
1D03 79         POSA2: MOV  A, C      ;LOAD ANTENNA NUMBER
1D04 FE14              CPI      20D   ;LAST ANTENNA?
1D06 C2DA1C            JNZ      POSA1 ;CHECK NEXT ANTENNA
                ;  LAST ANTENNA HAS BEEN CHECKED
1D09 78                MOV  A, B
1D0A A7                ANA  A         ;CHECK STORE FLAG
1D0B CA121D            JZ       NSTR1
                ;  IF ANTENNA 20 WAS A MAXIMUM VALUE,
                ;  CLEAR THE VALUE FOR THE NEXT ANTENNA
                ;  READING.
1D0E AF                XRA  A
1D0F 323B22            STA      ANEXT ;CLEAR NEXT ANTENNA VALUE
                ;  MAX ANTENNA HAS BEEN FOUND
                ;  MAXNO - ANTENNA NUMBER
                ;  APRES - A/D READING FOR THE MAX ANTENNA
                ;  ALAST - A/D READING FOR THE PREVIOUS ANTENNA
                ;  ANEXT - A/D READING FOR THE NEXT ANTENNA
1D12 CD561F     NSTR1: CALL     INCD5 ;POINT DE AT NEXT ANTENNA
1D15 D5                PUSH D         ;STORE DE REGISTER
                ;  LOAD NEARY WITH THE CLOSEST ANTENNA
1D16 3A3C22            LDA      MAXNO
1D19 323D22            STA      NEARY ;STORE NEAREST ANTENNA
                ;  MULTIPLY PRESENT VALUE BY 25
1D1C 3A3A22            LDA      APRES
1D1F 47                MOV  B, A
1D20 0E19              MVI  C, 25D    ;LOAD MULTIPLIER
1D22 CD701F            CALL     MULT8 ;APRES*25
                ;  NEGATE APRES*25
1D25 79                MOV  A, C
1D26 2F                CMA
1D27 6F                MOV  L, A
1D28 78                MOV  A, B
1D29 2F                CMA
1D2A 67                MOV  H, A
```

```
1D2B 23              INX  H          ;-(APRES*25)
1D2C E5              PUSH H
                ;  MULTIPLY ALAST BY MULTIPLIER CONSTANT
1D2D 3A3922          LDA      ALAST ;LOAD LAST ANTENNA VALUE
1D30 47              MOV  B, A
1D31 3A3F22          LDA      CMULT
1D34 4F              MOV  C, A       ;LOAD MULTIPLIER CONSTANT
1D35 CD701F          CALL     MULT8 ;ALAST * CMULT
                ;  SUBTRACT APRES*25
1D38 09              DAD  B          ;(ALAST*CMULT)-APRES*25
                ;  TEST RESULT OF SUBTRACTION
1D39 7C              MOV  A, H
1D3A A7              ANA  A          ;SET FLAGS
1D3B FA431D          JM       NCLOS ;NOT CLOSE
                ;  THE PREVIOUS ANTENNA READING IS CLOSE
                ;  TO THE MAXIMUM ANTENNA, SET THE
                ;  PREVIOUS CLOSE FLAG (REGISTER D).
1D3E 16FF            MVI  D, OFFH   ;SET PREVIOUS HIGH FLAG
1D40 C3451D          JMP      NCLOS+2
                ;  THE PREVIOUS ANTENNA READING WAS NOT
                ;  CLOSE TO THE MAXIMUM VALUE, CLEAR
                ;  THE PREVIOUS HIGH FLAG.
1D43 1600       NCLOS: MVI  D, 00H   ;CLEAR PREVIOUS HIGH FLAG
1D45 E1              POP  H          ;RESTORE -APRES*25 IN HL
                ;  MULTIPLY ANEXT BY THE MULTIPLIER CONSTANT
1D46 3A3B22          LDA      ANEXT ;LOAD NEXT ANTENNA DATA
1D49 47              MOV  B, A
1D4A 3A3F22          LDA      CMULT
1D4D 4F              MOV  C, A       ;LOAD MULTIPLIER CONSTANT
1D4E CD701F          CALL     MULT8 ;ALAST * CMULT
                ;  SUBTRACT APRES*25
1D51 09              DAD  B          ;(ALAST*CMULT)-APRES*25
                ;  TEST RESULT OF SUBTRACTION
1D52 7C              MOV  A, H
1D53 A7              ANA  A          ;SET FLAGS
1D54 FA5C1D          JM       NCLS1 ;NOT CLOSE
                ;  THE NEXT ANTENNA IS CLOSE TO THE
                ;  MAXIMUM VALUE, SET THE NEXT
                ;  HIGH FLAG (REGISTER E)>
1D57 1EFF            MVI  E, OFFH   ;SET HEXT HIGH FLAG
1D59 C35E1D          JMP      NCLS1+2
                ;   THE NEXT ANTENNA READING WAS NOT
                ;  CLOSE TO THE MAXIMUM VALUE, CLEAR
                ;  THE NEXT HIGH FLAG.
1D5C 1E00       NCLS1: MVI  E, 00H   ;CLEAR NEXT HIGH FLAG
                ;  READ PREVIOUS HIGH FLAG AND COMPARE
                ;  WITH THE HEXT HIGH FLAG
1D5E 7A              MOV  A, D       ;LOAD FLAG
1D5F BB              CMP  E          ;COMPARE LAST AND NEXT
1D60 0600            MVI  B, 00H     ;CLEAR HALF-WAY FLAG
1D62 CA731D          JZ       ONANT ;ANTENNA NUMBER IS CORRECT
                ;  TRANSMITTER IS BETWEEN ANTENNAS
1D65 06FF            MVI  B, OFFH   ;SET HALF-WAY FLAG
1D67 7B              MOV  A, E
```

```
1D68 A7              ANA  A          ;SET FLAGS
1D69 C2731D          JNZ      ONANT
               ;  PREVIOUS ANTENNA IS CLOSE, THERFORE
               ;  DECREMENT THE ANTENNA NUMBER
1D6C 3A3C22          LDA      MAXNO
1D6F 3D              DCR  A
1D70 323C22          STA      MAXNO ;DECREMENT ANTENNA NUMBER
               ;  TRANSMITTER IS CLOSEST TO THE MAXIMUM
               ;  ANTENNA, STORE THE ANTENNA NUMBER
1D73 3A3C22    ONANT: LDA     MAXNO ;LOAD ANTENNA NUMBER
1D76 CD731F          CALL     HXDEC ;CONVERT TO DECIMAL
               ;  STORE THE MS CHARACTER
1D79 4F              MOV  C, A       ;STORE IN C
1D7A E6F0            ANI  0F0H        ;MASK OFF LOW DIGIT
1D7C 0F              RRC
1D7D 0F              RRC
1D7E 0F              RRC
1D7F 0F              RRC             ;ROTATE INTO LOW NIBBLE
1D80 CD7C1F          CALL     CONAS ;CONVERT TO ASCII
1D83 32C821          STA      LCY5  ;STORE MS DIGIT
               ;  STORE THE LS CHARACTER
1D86 79              MOV  A, C       ;RELOAD DIGIT
1D87 E60F            ANI  0FH         ;MASK OFF MS DIGIT
1D89 CD7C1F          CALL     CONAS ;CONVERT TO ASCII
1D8C 32C921          STA      LCY5+1 ;STORE LS DIGIT
               ;  READ THE HALF WAY FLAG
1D8F 78              MOV  A, B
1D90 A7              ANA  A          ;READ HALF-WAY FLAG
1D91 CA9C1D          JZ       NOHF  ;NOT HALF-WAY
               ;  THE TRANSMITTER IS HALF WAY BETWEEN
               ;  ANTENNAS, STORE A FIVE FOR THE DECIMAL
               ;  PART OF THE ANTENNA NUMBER
1D94 3E35            MVI  A, '5'
1D96 32CA21          STA      LCY5+2 ;STORE A FIVE
1D99 C3A11D          JMP      LDED
               ;  THE TRANSMITTER IS NOT BETWEEN ANTENNAS,
               ;  STORE A ZERO AS THE DECIMAL PART
               ;  OF THE ANTENNA NUMBER
1D9C 3E30      NOHF:  MVI  A, '0'
1D9E 32CA21          STA      LCY5+2 ;STORE A ZERO
1DA1 D1        LDED:  POP  D
1DA2 C1              POP  B
1DA3 E1              POP  H
1DA4 C9              RET
               ;
               ;SUBROUTINE TO CALC. DIST. BETWEEN 2 TRANS.
               ;  POINT DE AT ONE SET OF DATA
               ;  POINT BC AT THE OTHER SET OF DATA
               ;  ANSWER IS RETURNED IN REG A
1DA5 E5        DIST:  PUSH H
               ;  PACK THE NUMBER POINTED TO BY BC
               ;  INTO ONE BYTE
1DA6 60              MOV  H, B
1DA7 69              MOV  L, C       ;MOVE BC TO HL
```

```
1DA8 CD031E              CALL    PACK  ;CREATE A DIGIT
                 ;  STORE THE X LOCATION
1DAB 324022              STA     AX    ;STORE X LOCATION
                 ;  PACK THE Y LOCATION
1DAE CD031E              CALL    PACK
1DB1 324122              STA     AY    ;STORE Y LOCATION
                 ;  PACK THE NUMBER POINTED TO BY DE
                 ;  INTO ONE BYTE
1DB4 62                  MOV  H, D
1DB5 6B                  MOV  L, E     ;MOVE DE TO HL
1DB6 CD031E              CALL    PACK  ;CREATE DIGIT
                 ;  STORE THE X LOCATION
1DB9 324222              STA     BX    ;STORE X LOCATION
                 ;  PACK THE Y LOCATION
1DBC CD031E              CALL    PACK
1DBF 324322              STA     BY    ;STORE Y LOCATION
                 ;  CALCULATE X AND Y DIMENSIONS
                 ;  AX AND AY CONTAIN THE COORDINATES
                 ;  FOR ONE TRANSMITTER
                 ;  BX AND BY CONTAIN THE COORDINATES
                 ;  FOR THE OTHER TRANSMITTER
1DC2 3A4022              LDA     AX
1DC5 214222              LXI  H, BX
1DC8 96                  SUB  M          ;AX-BX
1DC9 F2D31D              JP      DIST1
1DCC 3A4222              LDA     BX
1DCF 214022              LXI  H, AX
1DD2 96                  SUB  M          ;BX-AX
                 ;  REG A HAS THE X DIMENSION
1DD3 324022      DIST1:  STA     AX    ;STORE THE X DIMENSION
                 ; AX CONTAINS THE X DIMENSION
1DD6 3A4122              LDA     AY
1DD9 214322              LXI  H, BY
1DDC 96                  SUB  M          ;AY-BY
1DDD F2E71D              JP      DIST2
1DE0 3A4322              LDA     BY
1DE3 214122              LXI  H, AY
1DE6 96                  SUB  M          ;BY-AY
                 ;  REG A HAS THE Y DIMENSION
                 ;  SQUARE THE Y DIMENSION
1DE7 47          DIST2:  MOV  B, A
1DE8 4F                  MOV  C, A
1DE9 CD701F              CALL    MULT8 ;SQUARE Y DIMENSION
1DEC 50                  MOV  D, B
1DED 59                  MOV  E, C     ;MOVE TO DE REGISTER
                 ;  DE CONTAINS THE Y DIMENSION SQUARED
                 ;  SQUARE THE X DIMENSION
1DEE 3A4022              LDA     AX
1DF1 47                  MOV  B, A
1DF2 4F                  MOV  C, A
1DF3 CD701F              CALL    MULT8 ;SQUARE THE Y DIMENSION
1DF6 60                  MOV  H, B
1DF7 69                  MOV  L, C     ;MOVE TO HL REGISTER
                 ;  HL CONTAINS THE X DIMENSION SQUARED
```

```
1DF8 19              DAD  D        ;HL+DE
               ;  HL NOW CONTAINS (2X)**2 + (2X)**2
1DF9 54              MOV  D, H
1DFA 5D              MOV  E, L     ;MOVE TO THE DE REGISTER
               ;  DE CONTAINS THE SUM OF THE SQUARES
1DFB CD251E          CALL    SQRT
               ;  NOW DIVIDE BY TWO TO GET THE ANSWER
1DFE 37              STC
1DFF 3F              CMC           ;CLEAR CARRY
1E00 1F              RAR           ;DIVIDE BY 2
1E01 E1              POP  H
1E02 C9              RET
               ;
               ;SUBROUTINE TO PACK 3 DIGITS INTO ONE BYTE
               ;  HL POINTS TO THE START OF THE DATA TABLE
               ;  ANSWER IS RETURNED IN REGISTER A
               ;  ANSWER IS TWO TIMES THE NUMBER
               ;  I.E. IF HL POINTS AT THE CHARACTERS
               ;  1, 3 AND 5, THE ANSWER WILL BE 27
1E03 C5        PACK:  PUSH B
1E04 7E              MOV  A, M     ;LOAD MS DIGIT
1E05 CD791F          CALL    CONHX ;CONVERT TO DECIMAL
1E08 17              RAL
1E09 17              RAL
1E0A 17              RAL
1E0B 17              RAL           ;ROTATE INTO MS NIBBLE
1E0C 47              MOV  B, A     ;STORE IN REGISTER B
               ;  CONVERT LS DIGIT TO DECIMAL
1E0D 23              INX  H
1E0E 7E              MOV  A, M     ;LOAD LS DIGIT
1E0F CD791F          CALL    CONHX ;CONVERT TO DECIMAL
1E12 B0              ORA  B        ;ADD MS NIBBLE
1E13 CD761F          CALL    DECHX ;CONVERT TO HEX
               ;  A CONTAINS THE INTEGER PART OF
               ;  THE NUMBER
1E16 17              RAL           ;MULTIPLY BY 2
               ;  A CONTAINS 2 TIMES THE INTEGER
               ;  PART OF THE NUMBER
1E17 47              MOV  B, A     ;STORE IN REG B
1E18 23              INX  H        ;POINT AT FRACTIONAL PART
1E19 7E              MOV  A, M     ;LOAD DIGIT
1E1A FE30            CPI  '0'
1E1C 78              MOV  A, B     ;RESTORE A REGISTER
1E1D CA211E          JZ      PACK1
               ;  THERE IS A FRACTIONAL PART
1E20 3C              INR  A        ;ADD 1 IF FRACTION
1E21 23        PACK1: INX  H
1E22 23              INX  H
1E23 C1              POP  B
1E24 C9              RET
               ;
               ;SUBROUTINE TO CALCULATE SQUARE ROOTS
               ;  NUMBER IS ENTERED IN THE DE REGISTER
               ;  MAXIMUM VALUE IS 3F00H
```

```
                  ;  RESULT IS RETURNED IN THE A REGISTER
                  ;  THE SQUARE ROOT IS CALCULATED BY
                  ;  SUCCESSIVE APPROXIMATIONS.
                  ;  THE ROUTINE WILL KEEP TRYING NUMBERS
                  ;  UNTIL THE NEAREST INTEGER IS FOUND.
1E25 E5           SQRT:  PUSH H
1E26 C5                  PUSH B
1E27 267F                MVI  H, 7FH    ;LOAD MAX VALUE
1E29 2E00                MVI  L, 00H    ;LOAD MIN VALUE
1E2B F5           SQR1:  PUSH PSW
1E2C 7C                  MOV  A, H      ;LOAD MAX VALUE
1E2D 95                  SUB  L         ;SUBTRACT MIN VALUE
                  ;  IF H-L LEQ 1 EXIT FROM ROUTINE
1E2E CA5A1E              JZ      EXSQ
1E31 FE01                CPI  1
1E33 CA5A1E              JZ      EXSQ
1E36 33                  INX  SP
1E37 33                  INX  SP
                  ;  LOAD A WITH (H-L)/2 + L
1E38 1F                  RAR            ;DIVIDE BY 2
1E39 85                  ADD  L         ;ADD MIN VALUE
1E3A 47                  MOV  B, A
1E3B 4F                  MOV  C, A      ;LOAD BC REGISTER
1E3C CD701F              CALL    MULT8  ;SQUARE THE NUMBER
                  ;  BC NOW CONTAINS A SQUARED
1E3F F5                  PUSH PSW
1E40 E5                  PUSH H
                  ;  NEGATE BC AND MOVE TO HL
1E41 79                  MOV  A,C
1E42 2F                  CMA
1E43 6F                  MOV  L, A
1E44 78                  MOV  A, B
1E45 2F                  CMA
1E46 67                  MOV  H, A
1E47 23                  INX  H         ;NEGATE BC AND MOVE TO HL
                  ;  NUMBER - (GUESS)**2
1E48 19                  DAD  D         ;DE-BC
1E49 7C                  MOV  A, H
1E4A A7                  ANA  A         ;SET FLAGS
1E4B FA541E              JM      MINUS  ;BC>DE
                  ;  GUESS IS TOO LARGE, LOAD
                  ;  A NEW MINIMUM VALUE
1E4E E1                  POP  H
1E4F F1                  POP  PSW
1E50 6F                  MOV  L, A      ;LOAD NEW MIN VALUE
1E51 C32B1E              JMP     SQR1
                  ;  GUESS IS TOO SMALL, LOAD
                  ;  A NEW MAXIMUM VALUE
1E54 E1           MINUS: POP  H
1E55 F1                  POP  PSW
1E56 67                  MOV  H, A      ;LOAD NEW MAX VALUE
1E57 C32B1E              JMP     SQR1
1E5A F1           EXSQ:  POP  PSW       ;LOAD ANSWER
1E5B C1                  POP  B
```

```
1E5C E1              POP  H
1E5D C9              RET
               ;
               ;SUBROUTINE TO LOAD THE TEMPORARY PRINT
               ;  BUFFER INTO THE PRINT BUFFER.
1E5E C5        LDPRT: PUSH B
1E5F D5              PUSH D
1E60 E5              PUSH H
1E61 F5              PUSH PSW
1E62 2605            MVI  H, 5     ;LOAD COUNTER
1E64 016221          LXI  B, TM1   ;LOAD SOURCE ADDRESS
1E67 11D021          LXI  D, PRBF  ;LOAD DESTINATION ADDRESS
               ;  DECIDE WHETHER TO PRINT THE POSITIONS
               ;  OR THE FLAGS
1E6A 3A1A21          LDA      ACLOC
1E6D A7              ANA  A        ;SET FLAGS
1E6E C28C1E          JNZ      LDPT6 ;PRINT FLAGS
               ;  PRINT THE LOCATIONS
1E71 2E11      LDPT1: MVI  L, 17    ;LOAD BYTE COUNT
1E73 0A        LDPT2: LDAX B
1E74 12              STAX D        ;STORE DATA
1E75 03              INX  B
1E76 13              INX  D
1E77 2D              DCR  L
1E78 C2731E          JNZ      LDPT2 ;LINE NOT DONE
               ;  POSITIONS HAVE BEEN LOADED, LOAD
               ;  THE CARRIAGE CONTROL CHARACTER
1E7B 03              INX  B
1E7C 03              INX  B
1E7D 03              INX  B
1E7E 03              INX  B
1E7F 0A              LDAX B
1E80 12              STAX D        ;LOAD CARRIAGE CONTROL
1E81 03              INX  B
1E82 13              INX  D
1E83 25              DCR  H
1E84 C2711E          JNZ      LDPT1 ;NOT DONE YET
               ;  ALL FIVE LINES HAVE BEEN DONE
1E87 F1              POP  PSW
1E88 E1              POP  H
1E89 D1              POP  D
1E8A C1              POP  B
1E8B C9              RET
1E8C 2E0A      LDPT6: MVI  L, 10    ;LOAD BYTE COUNT
               ;  PRINT THE FLAGS
1E8E 0A        LDPT7: LDAX B
1E8F 12              STAX D        ;STORE DATA
1E90 03              INX  B
1E91 13              INX  D
1E92 2D              DCR  L
1E93 C28E1E          JNZ      LDPT7 ;LINE NOT DONE
               ;  TIME AND CHANNEL HAVE BEEN LOADED,
               ;  LOAD THE FLAGS
1E96 03              INX  B
```

```
1E97 03              INX  B
1E98 03              INX  B
1E99 03              INX  B
1E9A 03              INX  B
1E9B 03              INX  B
1E9C 03              INX  B
1E9D 03              INX  B
1E9E 2E04            MVI  L, 4
            ;  LOAD THE FLAGS AND CARRIAGE CONTROL
1EA0 0A     LDPT8:   LDAX B             ;STORE FLAG
1EA1 12              STAX D
1EA2 03              INX  B
1EA3 13              INX  D
1EA4 2D              DCR  L
1EA5 C2A01E          JNZ      LDPT8 ;LINE NOT DONE
1EA8 25              DCR  H
1EA9 C28C1E          JNZ      LDPT6 ;NEXT LINE
            ;  ALL FIVE LINES HAVE BEEN LOADED
1EAC F1              POP  PSW
1EAD E1              POP  H
1EAE D1              POP  D
1EAF C1              POP  B
1EB0 C9              RET
            ;
            ;SUBROUTINE TO LOAD A LINE FROM THE PRINT BUFF.
            ;  INTO THE CASSETTE BUFFER.
            ;  LINE NUMBER IS ENTERED IN REGISTER A
1EB1 C5     LDCAS:   PUSH B
1EB2 D5              PUSH D
1EB3 E5              PUSH H
1EB4 F5              PUSH PSW
1EB5 116221          LXI  D, TM1
1EB8 3D     LDCS1:   DCR  A             ;DECREMENT COUNTER
1EB9 CAC31E          JZ       LDCS2 ;DONE
1EBC CD671F          CALL     INDE  ;INCREMENT DE 21 TIMES
1EBF 13              INX  D             ;22 TIMES
1EC0 C3B81E          JMP      LDCS1
            ;  DE REGISTER NOW POINTS AT LINE TO BE OUTPUT
1EC3 2A2520 LDCS2:   LHLD     BUFER ;LOAD POINTER
            ;  LOAD GROUP SEPARATORS
1EC6 CD6D1F          CALL     NLINE
1EC9 0615            MVI  B, 21     ;LOAD COUNTER
            ;  LOAD DATA
1ECB 1A     LDCS3:   LDAX D             ;LOAD DATA
1ECC 77              MOV  M, A          ;STORE DATA
1ECD 23              INX  H
1ECE 13              INX  D
1ECF 05              DCR  B             ;DECREMENT COUNTER
1ED0 C2CB1E          JNZ      LDCS3 ;NOT DONE YET
            ;  LOAD CARRIAGE RETURN
1ED3 CD6A1F          CALL     CRLF  ;ENTER CARR. RETURN
1ED6 222520          SHLD     BUFER ;RESET BUFFER POINTER
1ED9 F1              POP  PSW
1EDA E1              POP  H
```

```
1EDB D1               POP  D
1EDC C1               POP  B
1EDD C9               RET
               ;
               ;SUBROUTINE TO INC. A SET OF POINTERS BY THE
               ;  VALUE IN REGISTER A.
               ;  REGISTERS TO BE INCREMENTED 1 TIME ARE
               ;  I1T1, I1T2, I1T3, I1T4, I1T5
               ;  REGISTERS TO BE INCREMENTED 22 TIMES ARE
               ;  I22T1, I22T2, I22T3
1EDE E5        INPNT: PUSH H
1EDF D5               PUSH D
1EE0 C5               PUSH B
1EE1 F5               PUSH PSW
1EE2 3D        INPT:  DCR  A
1EE3 CA101F           JZ        INP5  ;DONE
               ;  NOT DONE, INCREMENT THE POINTERS
1EE6 212F21           LXI  H, I1T1
1EE9 0605             MVI  B, 5       ;LOAD COUNTER
               ;  LOAD DE WITH DATA
1EEB 5E        INPT1: MOV  E, M
1EEC 23               INX  H
1EED 56               MOV  D, M       ;LOAD DE WITH DATA
1EEE 13               INX  D          ;INCREMENT
1EEF 72               MOV  M, D
1EF0 2B               DCX  H
1EF1 73               MOV  M, E       ;RESTORE DATA
               ;  POINTER HAS BEEN INCREMENTED BY 1
1EF2 23               INX  H
1EF3 23               INX  H
1EF4 05               DCR  B          ;DECREMENT COUNTER
1EF5 C2EB1E           JNZ       INPT1
               ;  ALL REGISTERS TO BE INCREMENTED
               ;  BY ONE ARE DONE.
               ;  POINT AT REGISTERS TO BE
               ;  INCREMENTED BY 22.
1EF8 213921           LXI  H, I22T1
1EFB 0603             MVI  B, 3       ;LOAD COUNTER
               ;  LOAD DE WITH DATA
1EFD 5E        INPT2: MOV  E, M
1EFE 23               INX  H
1EFF 56               MOV  D, M
1F00 CD671F           CALL      INDE  ;INCREMENT 21 TIMES
1F03 13               INX  D          ;22 TIMES
1F04 72               MOV  M, D
1F05 2B               DCX  H
1F06 73               MOV  M, E       ;RELOAD COUNTER
               ;  POINTER HAS BEEN INCREMENTED BY 22
1F07 23               INX  H
1F08 23               INX  H
1F09 05               DCR  B          ;DECREMENT COUNTER
1F0A C2FD1E           JNZ       INPT2
               ;  ALL PONTERS HAVE BEEN SERVICED
1F0D C3E21E           JMP       INPT
```

```
1F10 F1       INP5:  POP  PSW
1F11 C1              POP  B
1F12 D1              POP  D
1F13 E1              POP  H
1F14 C9              RET
              ;
              ;SUBROUTINE TO SET THE TEST ANTENNA ERROR FLAGS
              ;  THE SIGNAL FROM THE TEST ANTENNA IS
              ;  COMPARED WITH THE COMPARATOR VALUE.
              ;  IF THE SIGNAL IS TOO HIGH, THE FLAG
              ;  IS SET TO A ONE. OTHERWISE IT IS SET
              ;  TO A ZERO.
              ;  TANCP CONTAINS THE COMPARATOR VALUE
1F15 F5       STTAN: PUSH PSW
1F16 E5              PUSH H
1F17 C5              PUSH B
1F18 D5              PUSH D
1F19 210D21          LXI  H, TSTAN ;POINT AT ANTENNA VALUES
1F1C 117621          LXI  D, FLG1+2 ;POINT AT FIRST FLAG
1F1F 3A1221          LDA      TANCP
1F22 47              MOV  B, A     ;PUT COMPARATOR VALUE IN B
1F23 0E05            MVI  C, 5     ;LOAD COUNTER
              ;  REGISTER B CONTAINS THE COMPARATOR VALUE
1F25 7E       STT1:  MOV  A, M     ;LOAD ANTENNA VALUE
1F26 B8              CMP  B        ;COMPARE WITH SETPOINT
1F27 F2301F          JP       STEFL
              ;  CLEAR FLAG
1F2A 3E30            MVI  A, 30H
1F2C 12              STAX D        ;STORE A ZERO
1F2D C3331F          JMP      STT2
              ;  SET FLAG
1F30 3E31     STEFL: MVI  A, 31H
1F32 12              STAX D        ;STORE A ONE
1F33 CD671F   STT2:  CALL     INDE ;INCREMENT DE 21 TIMES
1F36 13              INX  D        ;22 TIMES
1F37 23              INX  H        ;POINT A NEXT CHANNEL
1F38 0D              DCR  C        ;DECREMENT COUNTER
1F39 C2251F          JNZ      STT1 ;NOT DONE YET
              ;  ALL FIVE FLAGS HAVE BEEN SERVICED
1F3C D1              POP  D
1F3D C1              POP  B
1F3E E1              POP  H
1F3F F1              POP  PSW
1F40 C9              RET
              ;
              ;SUBROUTINE TO SET THE ABOVE/BELOW FLAG
              ;  NOT IMPLEMENTED YET , THEREFORE STORE A ZERO
1F41 F5       STABL: PUSH PSW
1F42 E5              PUSH H
1F43 217521          LXI  H, FLG1+1
1F46 3E05            MVI  A, 5
1F48 3630     STAB1: MVI  M, 30H   ;STORE A ZERO
1F4A CD5C1F          CALL     IHL20 ;INCREMENT HL 20 TIMES
1F4D 23              INX  H
```

```
                                         ;22 TIMES
1F4E 23                  INX  H          ;DECREMENT COUNTER
1F4F 3D                  DCR  A
1F50 C2481F              JNZ       STAB1 ;NOT DONE YET
1F53 E1                  POP  H
1F54 F1                  POP  PSW
1F55 C9                  RET
                 ;
                 ;SUBROUTINE TO INCREMENT DE FIVE TIMES
1F56 13          INCD5:  INX  D
1F57 13                  INX  D
1F58 13                  INX  D
1F59 13                  INX  D
1F5A 13                  INX  D
1F5B C9                  RET
                 ;
                 ;SUBROUTINE TO INCREMENT HL 20 TIMES
1F5C CD611F      IHL20:  CALL      AD21
1F5F 2B                  DCX  H
1F60 C9                  RET
                 ;  FORWARDING ADDRESSES
1F61 C30300      AD21:   JMP       3
1F64 C30600      PRINT:  JMP       6
1F67 C30900      INDE:   JMP       9
1F6A C30C00      CRLF:   JMP       12
1F6D C30F00      NLINE:  JMP       15
1F70 C31200      MULT8:  JMP       18
1F73 C31500      HXDEC:  JMP       21
1F76 C31800      DECHX:  JMP       24
1F79 C31B00      CONHX:  JMP       27
1F7C C31E00      CONAS:  JMP       30
1F7F C32100      LEOF:   JMP       33
1F82 C32400      AD882:  JMP       36
                         END
```